
buildtest Documentation

Release 0.11.0

Shahzeb Siddiqui

Sep 09, 2021

BACKGROUND

1	Status	3
2	Useful Links	5
3	Description	7
3.1	Summary of buildtest	7
3.2	Installing buildtest	14
3.3	Buildtest Tutorial	17
3.4	Buildspec Tutorial	173
3.5	Configuring buildtest	285
3.6	Batch Scheduler Support	313
3.7	Build and Test Process	327
3.8	Additional Features	329
3.9	Buildtest Schemas	350
3.10	Using buildtest at HPC sites	422
3.11	Conference and Publications	423
3.12	Contributing Guide	424
3.13	API Reference	442
3.14	Buildtest Command Reference	516
4	License	533
5	Indices and tables	535
	Python Module Index	537
	Index	539

This documentation was last rebuild on Sep 09, 2021 and is intended for version 0.11.0.

If you are working off a latest release please see <https://buildtest.readthedocs.io/en/latest/> for documentation. If you are working off `devel` branch then please refer to <https://buildtest.readthedocs.io/en/devel/> which references the *devel* branch.

CHAPTER
ONE

STATUS

USEFUL LINKS

1. Source Code: <https://github.com/buildtesters/buildtest>
2. Documentation: <http://buildtest.rtf.d.io/>
3. Schema Docs: <https://buildtesters.github.io/buildtest/>
4. ReadTheDocs: <https://readthedocs.org/projects/buildtest/>
5. CodeCov: <https://codecov.io/gh/buildtesters/buildtest>
6. Slack Channel: <http://hpcbuildtest.slack.com>
7. Slack Invite: <https://hpcbuildtest.herokuapp.com>
8. CodeFactor: <https://www.codefactor.io/repository/github/buildtesters/buildtest>
9. Snyk: <https://app.snyk.io/org/buildtesters/>
10. Cori Test Repository: <https://github.com/buildtesters/buildtest-cori>

DESCRIPTION

`buildtest` is a testing framework to help HPC sites write test for their system as part of their routine acceptance & regression test. `buildtest` provides a YAML interface to write tests which `buildtest` can validate and generate shell scripts that can run on your HPC system. The test template (YAML) is called **buildspec** which can define one or more test instance that is validated by a `json schema`. `buildtest` supports the following batch schedulers: `IBM Spectrum LSF`, `Slurm`, `PBS` and `Cobalt`. We publish the schema documentation, json schemas, and schema examples at <https://buildtesters.github.io/buildtest/> which is useful when you are *writing buildspecs*.

To get started with `buildtest`, please see *installing buildtest* and *getting started guide*.

A spin-off project called `lmodule` is a Python API for `Lmod`. The `buildtest` module features were deprecated and moved to `lmodule` with the main objective is to automate module load testing. For more details on `lmodule` see <https://github.com/buildtesters/lmodule>

3.1 Summary of buildtest

Contents

- *Summary of buildtest*
 - *Background*
 - *Motivation*
 - *Inception of buildtest*
 - *Preview of buildtest*
 - * *Building Test*
 - * *Buildspec Interface*
 - * *Query Report*
 - * *Inspect Tests*
 - *Target Audience & Use Case*
 - *Timeline*
 - *Related Projects and community efforts*

3.1.1 Background

HPC System and Software Stack are tightly integrated with underlying architecture which makes them highly sensitive to changes in system such as OS, kernel, driver, or vendor updates. We need a testing framework to automate acceptance testing of an HPC system so that HPC Support Teams can increase **confidence** of their HPC system throughout the system lifecycle.

3.1.2 Motivation

There are many build automations tools for compiling source code into binary code, the most used tool is the **make** utility found in most Linux systems. Build scripts like **configure**, **cmake** and **autoconf** can generate files used by make for installing the software. Makefile is a file used by make program that shows how to compile and link a program which is the basis for building a software package. One can invoke **make test** which will run the target named **test** in Makefile that dictates how tests are compiled and run. Makefile is hard to interpret and requires in-depth experience with shell-scripting and strong understanding of how package is built and tested. Note that package maintainers must provide the source files, headers, and additional libraries to test the software and make test simply the test compilation and execution. Tools like *configure*, *cmake* and *autoconf* are insufficient for testing because HPC software stack consist of applications packaged in many formats and some are make-incompatible.

We wanted a framework that hides the complexity for compiling source code and provide an easy markup language to define test configuration to create the test. This leads to buildtest, which is a testing framework that generates test-scripts using YAML that is validated with JSON Schemas. YAML was picked given its ease-of-use and it lowers the barrier for writing tests.

3.1.3 Inception of buildtest

buildtest was founded by [Shahzeb Siddiqui](#) in 2017 when he was at [Pfizer](#) tasked for testing software stack for a data center migration.

Shahzeb was tasked with testing the software ecosystem by focusing on the most important application due to time constraints. During this period, several dozen test scripts were developed in shell-script that targeted core HPC tools such as compilers, **MPI**, **R**, **Python**, etc. A single master script was used to run all the tests which led to *buildtest*.

3.1.4 Preview of buildtest

You can run `buildtest help` followed by name of command and it will provide an overview of the buildtest.

Building Test

```
$ buildtest help build
```

Building Buildsspecs	

Command	Description
<code>buildtest build -b <file></code>	Build a single buildspec file
<code>buildtest build -b <dir></code>	Build all buildspecs.
<code>↪ recursively in a given directory</code>	
<code>buildtest build -b <file> -b <dir></code>	Build buildspecs by file and
<code>↪ directory</code>	

(continues on next page)

(continued from previous page)

buildtest build -b <file> -b <dir> -x <file> -x <dir> ↳when building buildsspecs	Exclude files and directory.
buildtest build -t pass -t python ↳'pass' and 'python'	Build buildsspecs by tagname
buildtest build -e <executor1> -e <executor2> ↳executor	Building buildsspecs by
buildtest build -b <file> -t <tagname1> -e <executor1> ↳file, directory, tags, and executors	Building buildsspecs with
buildtest build -b tutorials --filter type=script ↳'tutorials' and filter tests by type='script'	Build all tests in directory
buildtest build -b tutorials --filter tags=pass ↳'tutorials' and filter tests by tags='pass'	Build all tests in directory
buildtest build -b tutorials --filter maintainers=@bob ↳'tutorials' and filter tests by maintainers='@bob'	Build all tests in directory
buildtest build --helpfilter ↳used with --filter option	Show list of filter fields.
buildtest -c config.yml build -b <file> ↳file 'config.yml'	Use buildtest configuration.
buildtest build -b <file> --rebuild 5	Rebuild a test 5 times
buildtest build -b <file> --testdir /tmp	Write tests in /tmp

Buildspec Interface

```
$ buildtest help buildspec
```

Finding Buildsspecs

Command	Description
buildtest buildspec find ↳buildspecs and load all validated buildsspecs in cache	Discover and validate all.
buildtest buildspec find --rebuild	Rebuild cache file
buildtest buildspec find --root /tmp --rebuild ↳and rebuild buildspec cache	Discover buildsspecs in /tmp.
buildtest buildspec find --paths ↳for buildsspecs	Print all root directories.
buildtest buildspec find --buildspec ↳buildspecs from cache	List all available.
buildtest buildspec find --tags ↳cache	List all unique tags from.
buildtest buildspec find --executors ↳from cache	List all unique executors.
buildtest buildspec find --maintainers ↳cache	List all maintainers from.
buildtest buildspec find --maintainers-by-buildspecs ↳buildspecs by maintainer names.	Show breakdown of all.
buildtest buildspec find --filter type=script,tags=pass ↳on type=script and tags='pass'	Filter buildspec cache based.
buildtest buildspec find --filter buildspec=<path> ↳file	Filter cache by buildspec.

(continues on next page)

(continued from previous page)

buildtest buildspec find --format name,description ↪field: 'name' and 'description'	Format table columns by↵
buildtest buildspec find --group-by-tags	Group tests by tag name
buildtest buildspec find --group-by-executor	Group tests by executor name
buildtest buildspec find --helpfilter	Show all filter fields
buildtest buildspec find --helpformat	Show all format fields
buildtest buildspec find --terse ↪format	Display output in terse↵
buildtest buildspec find invalid	Show invalid buildspecs
buildtest buildspec find invalid --error ↪error messages	Show invalid buildspecs with↵
Validate buildspecs -----	
Command	Description
buildtest buildspec validate -b <file> ↪JSON Schema	Validate a buildspec with↵
buildtest buildspec validate -b /tmp/ -x /tmp/network ↪directory /tmp but exclude /tmp/network	Validate all buildspecs in↵
buildtest buildspec validate -t python -t mac ↪tagname 'python' and 'mac'	Validate all buildspecs for↵
buildtest buildspec validate -e generic.local.bash ↪executor 'generic.local.bash'	Validate all buildspecs for↵
Buildspec Summary -----	
Command	Description
buildtest buildspec summary ↪cache file	Show summary of buildspec↵
Show Content of buildspec -----	
Command	Description
buildtest buildspec show python_hello ↪based on test name 'python_hello'	Show content of buildspec↵

Query Report

```
$ buildtest help report
```

```
View Test Report
```

```
-----
```

Command	Description
<code>buildtest report</code>	Display all tests results
<code>buildtest report --filter returncode=0</code> ↳ <code>returncode=0</code>	Filter test results by
<code>buildtest report --filter state=PASS,tags=python</code> ↳ <code>filter fields.</code>	Filter test by multiple
<code>buildtest report --filter buildspec=tutorials/vars.yml</code> ↳ <code>file 'tutorials/vars.yml'</code>	Filter report by buildspec
<code>buildtest report --format name,state,buildspec</code> ↳ <code>'name', 'state', 'buildspec'</code>	Format report table by field
<code>buildtest report --helpfilter</code>	List all filter fields
<code>buildtest report --helpformat</code>	List all format fields
<code>buildtest report --oldest</code> ↳ <code>all tests</code>	Retrieve oldest record for
<code>buildtest report --latest</code> ↳ <code>all tests</code>	Retrieve latest record for
<code>buildtest report -r <report-file></code> ↳ <code>file to display test results</code>	Specify alternate report
<code>buildtest report --terse</code>	Print report in terse format
<code>buildtest report list</code>	List all report files
<code>buildtest report clear</code>	Remove content of report file
<code>buildtest report summary</code>	Show summary of test report

Inspect Tests

```
$ buildtest help inspect
```

```
Inspecting a Test
```

```
-----
```

Command	Description
<code>buildtest inspect list</code> ↳ <code>and corresponding builds spec file</code>	Display all test names, ids
<code>buildtest inspect list -t</code>	Show output in terse format
<code>buildtest inspect name hello</code>	Display all tests results
<code>buildtest inspect name foo bar</code> ↳ <code>'foo' and 'bar'</code>	Display record of test name
<code>buildtest inspect builds spec tutorials/vars.yml</code> ↳ <code>tests in builds spec file 'tutorials/vars.yml'</code>	Fetch latest runs for all
<code>buildtest inspect id <ID></code> ↳ <code>unique identifier</code>	Display record of test by

(continues on next page)

(continued from previous page)

<code>buildtest inspect query -o hello</code> ↪ file for test name 'hello'	Display content of output.
<code>buildtest inspect query -e hello</code> ↪ file for test name 'hello'	Display content of error.
<code>buildtest inspect query -d first -o -e foo bar</code> ↪ tests 'foo', 'bar', and show output and error file	Display first record of.
<code>buildtest inspect query -d all foo</code> ↪ 'foo'	Display all runs for tests

3.1.5 Target Audience & Use Case

buildtest target audience is *HPC Staff* that wants to perform acceptance & regression testing of their HPC system.

buildtest is not

- replacement for *make*, *cmake*, *autoconf*, *ctest*
- a software build framework (*easybuild*, *spack*, *nix* , *guix*)
- a replacement for benchmark tools or test suite from upstream package
- a replacement for writing tests, you will need to write your tests defined by buildtest schemas, however you can copy/paste & adapt tests from other sites that are applicable to you.

Typical use-case:

- Run your test suite during system maintenance
- Perform daily tests for testing various system components. These tests should be short
- Run weekly/biweekly test on medium/large workload including micro-benchmark
- Run tests for newly installed software package typically requested by user.

If you are interested trying out buildtest check out [Buildtest Tutorial](#) and [Join Slack Channel](#).

3.1.6 Timeline

Date	Version	Description
Sep 9th 2021	v0.11.0	Reimplement core implementation of running and polling jobs using asynchronous job submission. In addition we added several new commands including: buildtest cd , buildtest path and buildtest path and we enable alias for sub-commands.
Aug 16th 2021	v0.10.2	Add support for multi executor support in builds spec via <code>executors</code> property. Add new commands buildtest report summary for summary of report file. The buildtest builds spec show command shows content of builds spec file given a test name. The buildtest edit command can be used to edit builds spec and validate with JSON schema upon closing file. In this release, we added buildtest inspect builds spec command to view records based on builds spec file.
Jul 30th 2021	v0.10.1	Add new commands buildtest builds spec summary , buildtest builds spec invalid to show summary of builds spec cache and invalid builds specs. Add buildtest build --filter to filter builds specs during build. Add --terse option for several commands including buildtest history list , buildtest report , buildtest builds spec find . Add new command buildtest inspect query for querying test records. Added support for <code>metrics</code> property for defining arbitrary metrics in builds spec based on environment variable, variables or regular expression on stdout/stderr
Jul 13th 2021	v0.10.0	In this release we added spack support in buildtest by creating a new schema to write builds specs that will generate spack commands. For more details see <i>spack schema</i> . We added bash completion for buildtest commands which is enabled when installing buildtest. We added a new command buildtest builds spec validate that can be used for validating builds specs with JSON Schema.
Jun 11th 2021	v0.9.6	Added buildtest CDASH integration using buildtest cdash to upload test results. In this release we added buildtest history command to retrieve build history and query logfiles. Add global option -c in buildtest to specify alternate configuration file.
Mar 31th 2021	v0.9.5	Add support for PBS scheduler and reimplement buildtest inspect command
Mar 14th 2021	v0.9.4	Introduced major change in buildtest configuration file (<code>settings.schema.json</code>) to define multiple HPC systems in configuration file. This lead to change in how <code>executors</code> are referenced in builds spec file.
Feb 22nd 2021	v0.9.3	Change Copyright details for project to include LBNL . We added depend-abot for managing dependencies, added OLCF facility pipelines for running regression test.
Jan 12th 2021	v0.9.2	Contains major refactor to <code>compiler-v1.0-schema.json</code> for writing compiler test using regular expression to search for compilers that are defined in configuration file.
Nov 24st 2020	v0.9.1	Added support for Cobalt Scheduler .
Sep 3rd 2020	v0.8.0	Introduced JSON Schema for validating builds spec. Add support for Slurm and LSF scheduler for job submission. Add support for building builds specs by file, directory and tagname and command line interface to schema.
Sep 11th 2018	v0.4.0	buildtest was ported from Python 2 to 3.
Aug 20th 2017	v0.1.5	buildtest was converted from bash to Python and project was moved into github https://github.com/HPC-buildtest/buildtest .
Feb 18th 2017	N/A	Start of project

3.1.7 Related Projects and community efforts

Project	Description	State
ReFrame	is a high level regression framework for writing regression test for HPC systems. Tests are written in Python class and it has support for cray programming environment, job scheduler, module integration, parameter tests, test dependency, and sanity check. The project is led by CSCS .	Active
Pavilion2	is a framework for running and analyzing tests targeting HPC systems. Tests are written in YAML and majority of pavilion commands are implemented through python plugins using yapsy. Pavilion2 is developed by LANL .	Active
Automatic Testing of Installed Software (ATIS)	This project was presented by Xavier Besseron in FOSDEM14 that targets MPI testing using ctest and cdash. This project is no longer in development.	Obsolete
hpcswtest	is a HPC Software Stack Testing Framework developed by Idaho National Lab . The framework is built using C++11 and JSON file to define test configuration.	Obsolete
PVCS	is a validation engine to run large tests for HPC systems, the framework is written in Perl and recipe known as Test Expression (TE) are written in YAML. This project is developed by CEA .	Obsolete

The System Test Working Group hosted a BOF [HPC System Testing: Procedures, Acceptance, Regression Testing, and Automation](#) in SuperComputing '19. This working group is aimed at discussing acceptance and regression testing procedure and lessons learned from other HPC centers.

3.2 Installing buildtest

3.2.1 Requirements

You need the following packages to install buildtest.

- [git](#)
- [Python](#) >= 3.7

3.2.2 Cloning buildtest

To get started, clone the buildtest repository in your local machine as follows:

```
# HTTPS
$ git clone https://github.com/buildtesters/buildtest.git

# SSH
$ git clone git@github.com:buildtesters/buildtest.git
```

If you prefer the latest release use the **master** branch:

```
$ git clone -b master git@github.com:buildtesters/buildtest.git
```

3.2.3 Installing buildtest

buildtest requires a python 3.7 or higher, we recommend you setup a python environment in order to install buildtest. You can use [venv](#), [conda](#), or [pipenv](#) to manage your python environment depending on your preference.

venv Setup

```
python3 -m venv $HOME/buildtest
source $HOME/buildtest/activate
```

Conda Setup

```
conda create -n buildtest python=3.7
source activate buildtest
```

Pipenv Setup

```
pipenv --python 3.7
pipenv shell
```

Once you have your python environment setup, you can install buildtest, by sourcing the setup script depending on your shell type:

```
# BASH users
$ source setup.sh

# CSH users
$ source setup.csh
```

This will add `buildtest` command in your `$PATH` and set environment variable `$BUILDTEST_ROOT` which points to root of buildtest repo.

buildtest will provide tab completion for bash shell, this is managed by script `bash_completion.sh`, if you encounter any issues with tab completion please raise an issue at <https://github.com/buildtesters/buildtest/issues/>.

3.2.4 Development Dependencies (Optional)

If you plan to contribute back to buildtest, you will need to install additional dependencies as follows:

```
$ pip install -r docs/requirements.txt
```

3.2.5 Usage (buildtest --help)

Once you are setup, you can run `buildtest --help` for more details on how to use buildtest. Shown below is the output

```
$ buildtest --help
usage: buildtest [options] [COMMANDS]

buildtest is a HPC testing framework for building and running tests.

optional arguments:
  -h, --help            show this help message and exit
  -V, --version          show program's version number and exit
  -c CONFIGFILE, --config CONFIGFILE
                        Specify Path to Configuration File
  -d, --debug           Print debug messages to screen
  --color {on,off}      Enable or disable color

COMMANDS:

  build (bd)            Build and Run test
  buildspect (bc)       Buildspect Interface
  config (cg)           Query buildtest configuration
  report (rt)           Query test report
  inspect (it)          Inspect a test based on NAME or ID
  history (hy)          Query build history
  edit (et)             Edit a buildspect and validate with schema file
  schema               List schema contents and examples
  cdash                Upload test to CDASH server
  cd                   change directory to root of test given a test name
  clean                Remove all generate files from buildtest including test_
↳ directory, log files, report file,
                        buildspect cache, history files.
  path                 Show path attributes for a given test
  docs                 Open buildtest docs in browser
  schemadocs           Open buildtest schema docs in browser
  help (h)             buildtest command guide
```

References

GitHub:	https://github.com/buildtesters/buildtest
Documentation:	https://buildtest.readthedocs.io/en/latest/index.html
Schema Documentation:	https://buildtesters.github.io/buildtest/
Slack:	http://hpcbuildtest.slack.com/

Please report issues at <https://github.com/buildtesters/buildtest/issues>

Copyright (c) 2021, The Regents of the University of California, through Lawrence_↳
Berkeley National Laboratory (subject to receipt of any required approvals from the U.
↳S. Dept. of Energy), Shahzeb Siddiqui, and Vanessa Sochat. All rights reserved.

If you have got this far, please go to the next section on *Buildtest Tutorial*

3.3 Buildtest Tutorial

3.3.1 Building Test via buildtest (buildtest build)

This guide will get you familiar with buildtest command line interface. Once you complete this section, you can proceed to *writing buildspects* section where we will cover how to write buildspects.

Once you install buildtest, you should find the *buildtest* command in your **\$PATH**. You can check the path to buildtest command by running:

```
$ which buildtest
```

If you don't see buildtest go back and *install buildtest*.

When you clone buildtest, you also get a set of buildspects that you can run on your system. The *buildtest build* command is used for building and running tests. Buildtest will read one or more buildspects file that adheres to one of the buildtest schemas. For a complete list of build options please run *buildtest build --help*.

Note: *buildtest bd* is an alias for *buildtest build* command.

Build Usage

```
$ buildtest build --help
usage: buildtest [options] [COMMANDS] build [-h] [-b BUILDSPEC] [-x EXCLUDE] [-e EXECUTOR] [-t TAGS] [-f FILTER]
      [--helpfilter] [--disable-executor-check] [--max-pend-time MAX_PEND_TIME] [--poll-interval POLL_INTERVAL]
      [--rebuild REBUILD] [-r REPORT] [--retry RETRY] [-s {parse,build}] [--testdir TESTDIR]

optional arguments:
  -h, --help                show this help message and exit

discover:
  select buildspects

  -b BUILDSPEC, --buildspec BUILDSPEC
                        Specify a buildspect (file or directory) to build. A buildspect must end in '.yaml' extension.
  -x EXCLUDE, --exclude EXCLUDE
                        Exclude one or more buildspects (file or directory) from processing. A buildspect must end in '.yaml' extension.
  -e EXECUTOR, --executor EXECUTOR
                        Discover buildspects by executor name found in buildspect cache
  -t TAGS, --tags TAGS     Discover buildspects by tags found in buildspect cache

filter:
```

(continues on next page)

(continued from previous page)

```

Filter tests

-f FILTER, --filter FILTER
                        Filter buildspec based on tags, type, or maintainers. Usage: --
↪ filter key1=val1,key2=val2
--helpfilter           Show available filter fields used with --filter option

extra:
  All extra options

--disable-executor-check
                        Disable executor check during configuration check. By default,
↪ these checks are enforced for
                        Local, Slurm, PBS, LSF, and Cobalt Executor.
-k, --keep-stage-dir  Keep stage directory after job completion.
--max-pend-time MAX_PEND_TIME
                        Specify Maximum Pending Time (sec) for job before cancelling job.
↪ This only applies for batch
                        job submission.
--poll-interval POLL_INTERVAL
                        Specify Poll Interval (sec) for polling batch jobs
--rebuild REBUILD      Rebuild test X number of times. Must be a positive number,
↪ between [1-50]
-r REPORT, --report REPORT
                        Specify a report file where tests will be written.
--retry RETRY          Retry failed jobs
-s {parse,build}, --stage {parse,build}
                        control behavior of buildtest build
--testdir TESTDIR      Specify a custom test directory where to write tests. This,
↪ overrides configuration file and
                        default location.

```

Building a Test

To build a test, we use the `--buildspec` or short option `-b` to specify the path to buildspec file. Let's see some examples, first we specify a full path to buildspec file. In this example, buildtest will *discover buildspects* followed by parsing the test with appropriate schema and generate a shell script that is run by buildtest. You can learn more about *build and test process*.

```

$ buildtest build -b $BUILDTEST_ROOT/tutorials/vars.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:54:50
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↪ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests

```

(continues on next page)

(continued from previous page)

```

Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/vars.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml: VALID

Total builder objects created: 1
builders: [variables_bash/98b07db8]

name          id          description          buildsspecs
-----
↳ -----
variables_bash 98b07db8 Declare shell variables in bash /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type  | executor          | tags          | testpath
-----+-----+-----+-----+-----+-----
↳ -----
↳ -----
variables_bash | 98b07db8 | script | generic.local.bash | ['tutorials'] | /home/docs/
↳ checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic
↳ local.bash/vars/variables_bash/98b07db8/variables_bash_build.sh

```

(continued from previous page)

```
+-----+
| Stage: Running Test |
+-----+

variables_bash/98b07db8: completed with returncode: 0
variables_bash/98b07db8: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↳ 98b07db8/variables_bash.out
variables_bash/98b07db8: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↳ 98b07db8/variables_bash.err

-----
Launching test: variables_bash
Test ID: 98b07db8-9b3e-4ff0-b526-1ebcebbc3a1e
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash_build.sh

+-----+
| Stage: Test Summary |
+-----+

name          | id          | executor          | status | returncode_match | regex_
↳ match      | runtime_match | returncode | runtime
-----+-----+-----+-----+-----+-----+
↳ -----+-----+-----+-----+-----+
variables_bash | 98b07db8 | generic.local.bash | PASS   | N/A              | N/A
↳           | N/A      |           0 | 0.012987

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_8xn_louq.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log
```

Note: buildtest will only read buildsspecs with .yaml extension, if you specify a .yaml it will be ignored by buildtest.

The --buildspec option can be used to specify a file or directory path. If you want to build multiple buildsspecs in a directory you can specify the directory path and buildtest will recursively search for all .yaml files. In the next example, we build all tests in directory **general_tests/configuration**.

```
$ buildtest build -b general_tests/configuration/
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:12
```

(continues on next page)

(continued from previous page)

```

buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳bin/buildtest build -b general_tests/configuration/

```

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

```

```

+-----+
↳-----+
| Discovered Buildsspecs |
↳
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳tests/configuration/ulimits.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳tests/configuration/ssh_localhost.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳tests/configuration/kernel_state.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳tests/configuration/disk_usage.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳tests/configuration/systemd-default-target.yml |
+-----+
↳-----+

```

```

Discovered Buildsspecs: 5
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 5

```

```

+-----+
| Stage: Parsing Buildsspecs |
+-----+

```

```

Valid Buildsspecs: 5
Invalid Buildsspecs: 0

```

(continues on next page)

(continued from previous page)

```

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ulimits.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ssh_localhost.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/kernel_state.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/disk_usage.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/systemd-default-target.yml: VALID

```

Total builder objects created: 10

```

builders: [ulimit_filelock_unlimited/b1b5ebdc, ulimit_cputime_unlimited/aa16f2ae, ulimit_
↳ stacksize_unlimited/7204f37a, ulimit_vmsize_unlimited/247638c0, ulimit_filedescriptor_
↳ 4096/677a3ea5, ulimit_max_user_process_2048/81105021, ssh_localhost_remotecommand/
↳ d99f5389, kernel_swapusage/c6406429, root_disk_usage/6b5e2ea2, systemd_default_target/
↳ daaa00e7]

```

name	id	description
↳ buildspecs		

↳ -----		
ulimit_filelock_unlimited	b1b5ebdc	Check if file lock is set to unlimited in
↳ ulimits		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml		
ulimit_cputime_unlimited	aa16f2ae	Check if cputime is set to unlimited in ulimits
↳		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/ulimits.yml		
ulimit_stacksize_unlimited	7204f37a	Check if stack size is set to unlimited in
↳ ulimits		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml		
ulimit_vmsize_unlimited	247638c0	Check virtual memory size and check if its set
↳ to unlimited		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml		
ulimit_filedescriptor_4096	677a3ea5	Check if open file descriptors limit is set to
↳ 4096		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml		
ulimit_max_user_process_2048	81105021	Check max number of user process limit is set to
↳ 2048		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/ulimits.yml		
ssh_localhost_remotecommand	d99f5389	Test if ssh on localhost works and if we can run
↳ remote command.		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/ssh_localhost.yml		
kernel_swapusage	c6406429	Retrieve Kernel Swap Usage
↳		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/kernel_state.yml		
root_disk_usage	6b5e2ea2	Check root disk usage and report if it exceeds
↳ threshold		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/disk_usage.yml		

(continues on next page)

(continued from previous page)

```

systemd_default_target      daaa00e7  check if default target is multi-user.target
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/systemd-default-target.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags
↳            | testpath
-----+-----+-----+-----+-----+
↳
↳
↳
ulimit_filelock_unlimited    | b1b5ebdc | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_filelock_unlimited/b1b5ebdc/ulimit_
↳ filelock_unlimited_build.sh
ulimit_cputime_unlimited     | aa16f2ae | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_cputime_unlimited/aa16f2ae/ulimit_
↳ cputime_unlimited_build.sh
ulimit_stacksize_unlimited   | 7204f37a | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_stacksize_unlimited/7204f37a/ulimit_
↳ stacksize_unlimited_build.sh
ulimit_vmsize_unlimited      | 247638c0 | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_vmsize_unlimited/247638c0/ulimit_
↳ vmsize_unlimited_build.sh
ulimit_filedescriptor_4096   | 677a3ea5 | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_filedescriptor_4096/677a3ea5/ulimit_
↳ filedescriptor_4096_build.sh
ulimit_max_user_process_2048 | 81105021 | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_max_user_process_2048/81105021/ulimit_
↳ max_user_process_2048_build.sh
ssh_localhost_remotecommand | d99f5389 | script | generic.local.bash | ['ssh']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ssh_localhost/ssh_localhost_remotecommand/d99f5389/ssh_
↳ localhost_remotecommand_build.sh
kernel_swapusage            | c6406429 | script | generic.local.bash | ['configuration
↳ '] /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/kernel_state/kernel_swapusage/c6406429/kernel_
↳ swapusage_build.sh
root_disk_usage              | 6b5ebea2 | script | generic.local.bash | ['filesystem',
↳ 'storage'] /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/disk_usage/root_disk_usage/6b5ebea2/root_disk_usage_
↳ build.sh
systemd_default_target       | daaa00e7 | script | generic.local.bash | ['system']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/systemd-default-target/systemd_default_target/daaa00e7/
↳ systemd_default_target_build.sh

```

(continued from previous page)

```
+-----+
| Stage: Running Test |
+-----+

ulimit_filelock_unlimited/b1b5ebdc: completed with returncode: 0
ulimit_filelock_unlimited/b1b5ebdc: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_filelock_unlimited/b1b5ebdc/ulimit_filelock_unlimited.out
ulimit_filelock_unlimited/b1b5ebdc: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_filelock_unlimited/b1b5ebdc/ulimit_filelock_unlimited.err
ulimit_filelock_unlimited/b1b5ebdc: performing regular expression - '^unlimited$' on
↳file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳tests/generic.local.bash/ulimits/ulimit_filelock_unlimited/b1b5ebdc/ulimit_filelock_
↳unlimited.out with regular expression
ulimit_filelock_unlimited/b1b5ebdc: Regular Expression Match - Success!
ulimit_stacksize_unlimited/7204f37a: completed with returncode: 0
ulimit_stacksize_unlimited/7204f37a: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_stacksize_unlimited/7204f37a/ulimit_stacksize_unlimited.out
ulimit_stacksize_unlimited/7204f37a: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_stacksize_unlimited/7204f37a/ulimit_stacksize_unlimited.err
ulimit_stacksize_unlimited/7204f37a: performing regular expression - '^unlimited$' on
↳file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳tests/generic.local.bash/ulimits/ulimit_stacksize_unlimited/7204f37a/ulimit_stacksize_
↳unlimited.out with regular expression
ulimit_stacksize_unlimited/7204f37a: Regular Expression Match - Failed!
ulimit_filedescriptor_4096/677a3ea5: completed with returncode: 0
ulimit_filedescriptor_4096/677a3ea5: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_filedescriptor_4096/677a3ea5/ulimit_filedescriptor_4096.out
ulimit_filedescriptor_4096/677a3ea5: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_filedescriptor_4096/677a3ea5/ulimit_filedescriptor_4096.err
ulimit_filedescriptor_4096/677a3ea5: performing regular expression - '^4096$' on file: /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/ulimits/ulimit_filedescriptor_4096/677a3ea5/ulimit_filedescriptor_
↳4096.out with regular expression
ulimit_filedescriptor_4096/677a3ea5: Regular Expression Match - Failed!
ssh_localhost_remotecommand/d99f5389: completed with returncode: 255
ssh_localhost_remotecommand/d99f5389: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ssh_localhost/ssh_localhost_remotecommand/d99f5389/ssh_localhost_remotecommand.out
ssh_localhost_remotecommand/d99f5389: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ssh_localhost/ssh_localhost_remotecommand/d99f5389/ssh_localhost_remotecommand.err
ulimit_cputime_unlimited/aa16f2ae: completed with returncode: 0
ulimit_cputime_unlimited/aa16f2ae: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_cputime_unlimited/aa16f2ae/ulimit_cputime_unlimited.out
```

(continues on next page)

(continued from previous page)

```

ulimit_cputime_unlimited/aa16f2ae: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_cputime_unlimited/aa16f2ae/ulimit_cputime_unlimited.err
ulimit_cputime_unlimited/aa16f2ae: performing regular expression - '^unlimited$' on
↳file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳tests/generic.local.bash/ulimits/ulimit_cputime_unlimited/aa16f2ae/ulimit_cputime_
↳unlimited.out with regular expression
ulimit_cputime_unlimited/aa16f2ae: Regular Expression Match - Success!
ulimit_vmsize_unlimited/247638c0: completed with returncode: 0
ulimit_vmsize_unlimited/247638c0: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_vmsize_unlimited/247638c0/ulimit_vmsize_unlimited.out
ulimit_vmsize_unlimited/247638c0: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_vmsize_unlimited/247638c0/ulimit_vmsize_unlimited.err
ulimit_vmsize_unlimited/247638c0: performing regular expression - '^unlimited$' on file:
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/ulimits/ulimit_vmsize_unlimited/247638c0/ulimit_vmsize_unlimited.
↳out with regular expression
ulimit_vmsize_unlimited/247638c0: Regular Expression Match - Success!
ulimit_max_user_process_2048/81105021: completed with returncode: 0
ulimit_max_user_process_2048/81105021: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_max_user_process_2048/81105021/ulimit_max_user_process_2048.out
ulimit_max_user_process_2048/81105021: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_max_user_process_2048/81105021/ulimit_max_user_process_2048.err
ulimit_max_user_process_2048/81105021: performing regular expression - '^2048$' on file:
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/ulimits/ulimit_max_user_process_2048/81105021/ulimit_max_user_
↳process_2048.out with regular expression
ulimit_max_user_process_2048/81105021: Regular Expression Match - Failed!
kernel_swapusage/c6406429: completed with returncode: 127
kernel_swapusage/c6406429: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/
↳kernel_swapusage/c6406429/kernel_swapusage.out
kernel_swapusage/c6406429: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/kernel_
↳swapusage/c6406429/kernel_swapusage.err
root_disk_usage/6b5ebea2: completed with returncode: 0
root_disk_usage/6b5ebea2: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳usage/6b5ebea2/root_disk_usage.out
root_disk_usage/6b5ebea2: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳usage/6b5ebea2/root_disk_usage.err
systemd_default_target/daaa00e7: completed with returncode: 1
systemd_default_target/daaa00e7: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-
↳default-target/systemd_default_target/daaa00e7/systemd_default_target.out
systemd_default_target/daaa00e7: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-
↳default-target/systemd_default_target/daaa00e7/systemd_default_target.err

```

(continues on next page)

(continued from previous page)

```

-----
Launching test: ulimit_filelock_unlimited
Test ID: b1b5ebdc-dc46-4979-a4dc-6087543fc867
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_filelock_unlimited/b1b5ebdc/ulimit_
↳filelock_unlimited_build.sh

```

```

-----
Launching test: ulimit_cputime_unlimited
Test ID: aa16f2ae-4389-4b94-b5b6-0e4a3635352e
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_cputime_unlimited/aa16f2ae/ulimit_
↳cputime_unlimited_build.sh

```

```

-----
Launching test: ulimit_stacksize_unlimited
Test ID: 7204f37a-7229-4b7d-9c7a-81fc31f0e6c0
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_stacksize_unlimited/7204f37a/ulimit_
↳stacksize_unlimited_build.sh

```

```

-----
Launching test: ulimit_vmsize_unlimited
Test ID: 247638c0-e13f-4489-8197-e65b56ea94db
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_vmsize_unlimited/247638c0/ulimit_
↳vmsize_unlimited_build.sh

```

```

-----
Launching test: ulimit_filedescriptor_4096
Test ID: 677a3ea5-d4ee-492f-8aca-24c59d7eefb9
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_filedescriptor_4096/677a3ea5/ulimit_
↳filedescriptor_4096_build.sh

```

```

-----
Launching test: ulimit_max_user_process_2048
Test ID: 81105021-047b-4ed3-b73f-676c7b086dbd
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_max_user_process_2048/81105021/ulimit_
↳max_user_process_2048_build.sh

```

```

-----
Launching test: ssh_localhost_remotecommand
Test ID: d99f5389-93a9-49fd-95ed-695e1b733e08
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ssh_localhost/ssh_localhost_remotecommand/d99f5389/
↳ssh_localhost_remotecommand_build.sh

```

```

-----
Launching test: kernel_swapusage
Test ID: c6406429-ff5c-49b2-9315-d5bcfa2e3be7

```

(continues on next page)

(continued from previous page)

Executor Name: generic.local.bash
 Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
 ↳ 11.0/var/tests/generic.local.bash/kernel_state/kernel_swapusage/c6406429/kernel_
 ↳ swapusage_build.sh

 Launching test: root_disk_usage

Test ID: 6b5ebea2-775e-4d52-bebe-7780a146a31d

Executor Name: generic.local.bash

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
 ↳ 11.0/var/tests/generic.local.bash/disk_usage/root_disk_usage/6b5ebea2/root_disk_usage_
 ↳ build.sh

 Launching test: systemd_default_target

Test ID: daaa00e7-1225-4a6f-ae2d-741234477d87

Executor Name: generic.local.bash

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
 ↳ 11.0/var/tests/generic.local.bash/systemd-default-target/systemd_default_target/
 ↳ daaa00e7/systemd_default_target_build.sh

+-----+
 | Stage: Test Summary |
 +-----+

name	id	executor	status	returncode_
↳ match regex_match runtime_match		↳ returncode	↳ runtime	
-----+-----+-----+-----+-----				
ulimit_filelock_unlimited	b1b5ebdc	generic.local.bash	PASS	False
↳ True False		0 0.022418		↳
ulimit_cputime_unlimited	aa16f2ae	generic.local.bash	PASS	False
↳ True False		0 0.01952		↳
ulimit_stacksize_unlimited	7204f37a	generic.local.bash	FAIL	False
↳ False False		0 0.023393		↳
ulimit_vmsize_unlimited	247638c0	generic.local.bash	PASS	False
↳ True False		0 0.024396		↳
ulimit_filedescriptor_4096	677a3ea5	generic.local.bash	FAIL	False
↳ False False		0 0.008116		↳
ulimit_max_user_process_2048	81105021	generic.local.bash	FAIL	False
↳ False False		0 0.021057		↳
kernel_swapusage	c6406429	generic.local.bash	FAIL	N/A
↳ N/A N/A		127 0.020363		↳
root_disk_usage	6b5ebea2	generic.local.bash	PASS	N/A
↳ N/A N/A		0 0.039046		↳
systemd_default_target	daaa00e7	generic.local.bash	FAIL	N/A
↳ N/A N/A		1 0.040417		↳
ssh_localhost_remotecommand	d99f5389	generic.local.bash	FAIL	N/A
↳ N/A N/A		255 0.155613		↳

Passed Tests: 4/10 Percentage: 40.000%

Failed Tests: 6/10 Percentage: 60.000%

(continues on next page)

(continued from previous page)

```
Writing Logfile to: /tmp/buildtest_jpcua50c.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log
```

Building Multiple Buildsspecs

You can append `-b` option to build multiple buildsspecs in the same command. Buildtest will discover buildsspecs for every argument (`-b`) and accumulate a list of buildsspecs to run. In this example, we instruct buildtest to build a buildspec file and all buildsspecs in a directory path.

```
$ buildtest build -b general_tests/configuration/ -b tutorials/vars.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:13
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b general_tests/configuration/ -b tutorials/vars.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ | Discovered Buildsspecs |
+-----+
=====
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ssh_localhost.yml |
+-----+
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/kernel_state.yml |
+-----+
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ulimits.yml |
+-----+
↳
```

(continues on next page)

(continued from previous page)

```

| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/systemd-default-target.yml |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/disk_usage.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 6
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 6

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 6
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ssh_localhost.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/kernel_state.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ulimits.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/systemd-default-target.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/disk_usage.yml: VALID

Total builder objects created: 11
builders: [ssh_localhost_remotecommand/93086006, kernel_swapusage/5ef212a6, ulimit_
↳ filelock_unlimited/ac5c8bd1, ulimit_cputime_unlimited/3914f699, ulimit_stacksize_
↳ unlimited/f2782a7c, ulimit_vmsize_unlimited/55aecca, ulimit_filedescriptor_4096/
↳ fe1869b1, ulimit_max_user_process_2048/7d195a72, systemd_default_target/d016e48c,
↳ variables_bash/7ff48178, root_disk_usage/edf300cd]

name                               id          description
↳ buildsspecs
-----
↳ -----
↳ -----
ssh_localhost_remotecommand 93086006 Test if ssh on localhost works and if we can run
↳ remote command. /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/ssh_localhost.yml

```

(continues on next page)

(continued from previous page)

```

kernel_swapusage      5ef212a6  Retrieve Kernel Swap Usage
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/kernel_state.yml
ulimit_filelock_unlimited  ac5c8bd1  Check if file lock is set to unlimited in
↳ ulimits /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml
ulimit_cputime_unlimited  3914f699  Check if cputime is set to unlimited in ulimits
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/ulimits.yml
ulimit_stacksize_unlimited  f2782a7c  Check if stack size is set to unlimited in
↳ ulimits /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml
ulimit_vm_size_unlimited  55aecca  Check virtual memory size and check if its set
↳ to unlimited /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml
ulimit_filedescriptor_4096  fe1869b1  Check if open file descriptors limit is set to
↳ 4096 /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/ulimits.yml
ulimit_max_user_process_2048  7d195a72  Check max number of user process limit is set to
↳ 2048 /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/ulimits.yml
systemd_default_target  d016e48c  check if default target is multi-user.target
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/configuration/systemd-default-target.yml
variables_bash          7ff48178  Declare shell variables in bash
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/tutorials/vars.yml
root_disk_usage         edf300cd  Check root disk usage and report if it exceeds
↳ threshold /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/general_tests/configuration/disk_usage.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags
↳ | testpath
-----+-----+-----+-----+-----
↳ +-----+
↳ -----
↳ -----
ssh_localhost_remotecommand | 93086006 | script | generic.local.bash | ['ssh']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ssh_localhost/ssh_localhost_remotecommand/93086006/ssh_
↳ localhost_remotecommand_build.sh
kernel_swapusage | 5ef212a6 | script | generic.local.bash | ['configuration
↳ '] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/kernel_state/kernel_swapusage/5ef212a6/kernel_
↳ swapusage_build.sh
ulimit_filelock_unlimited | ac5c8bd1 | script | generic.local.bash | ['system']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ulimits/ulimit_filelock_unlimited/ac5c8bd1/ulimit_
↳ filelock_unlimited_build.sh

```

(continues on next page)

(continued from previous page)

```

ulimit_cputime_unlimited      | 3914f699 | script | generic.local.bash | ['system']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/ulimits/ulimit_cputime_unlimited/3914f699/ulimit_
→cputime_unlimited_build.sh
ulimit_stacksize_unlimited   | f2782a7c | script | generic.local.bash | ['system']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/ulimits/ulimit_stacksize_unlimited/f2782a7c/ulimit_
→stacksize_unlimited_build.sh
ulimit_vmsize_unlimited      | 55aecca | script | generic.local.bash | ['system']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/ulimits/ulimit_vmsize_unlimited/55aecca/ulimit_
→vmsize_unlimited_build.sh
ulimit_filedescriptor_4096  | fe1869b1 | script | generic.local.bash | ['system']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/ulimits/ulimit_filedescriptor_4096/fe1869b1/ulimit_
→filedescriptor_4096_build.sh
ulimit_max_user_process_2048 | 7d195a72 | script | generic.local.bash | ['system']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/ulimits/ulimit_max_user_process_2048/7d195a72/ulimit_
→max_user_process_2048_build.sh
systemd_default_target     | d016e48c | script | generic.local.bash | ['system']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/systemd-default-target/systemd_default_target/d016e48c/
→systemd_default_target_build.sh
variables_bash              | 7ff48178 | script | generic.local.bash | ['tutorials']
→ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash_build.sh
root_disk_usage             | edf300cd | script | generic.local.bash | ['filesystem',
→ 'storage'] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
→11.0/var/tests/generic.local.bash/disk_usage/root_disk_usage/edf300cd/root_disk_usage_
→build.sh

+-----+
| Stage: Running Test |
+-----+

kernel_swapusage/5ef212a6: completed with returncode: 127
kernel_swapusage/5ef212a6: Writing output file: /home/docs/checkouts/readthedocs.org/
→user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/
→kernel_swapusage/5ef212a6/kernel_swapusage.out
kernel_swapusage/5ef212a6: Writing error file: /home/docs/checkouts/readthedocs.org/user_
→builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/kernel_
→swapusage/5ef212a6/kernel_swapusage.err
ulimit_filelock_unlimited/ac5c8bd1: completed with returncode: 0
ulimit_filelock_unlimited/ac5c8bd1: Writing output file: /home/docs/checkouts/
→readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
→ulimits/ulimit_filelock_unlimited/ac5c8bd1/ulimit_filelock_unlimited.out
ulimit_filelock_unlimited/ac5c8bd1: Writing error file: /home/docs/checkouts/readthedocs.
→org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
→ulimit_filelock_unlimited/ac5c8bd1/ulimit_filelock_unlimited.err
ulimit_filelock_unlimited/ac5c8bd1: performing regular expression - '^unlimited$' on
→file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
→tests/generic.local.bash/ulimits/ulimit_filelock_unlimited/ac5c8bd1/ulimit_filelock_
→unlimited.out with regular expression

```

(continued from previous page)

```

ulimit_filelock_unlimited/ac5c8bd1: Regular Expression Match - Success!
ulimit_stacksize_unlimited/f2782a7c: completed with returncode: 0
ulimit_stacksize_unlimited/f2782a7c: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_stacksize_unlimited/f2782a7c/ulimit_stacksize_unlimited.out
ulimit_stacksize_unlimited/f2782a7c: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_stacksize_unlimited/f2782a7c/ulimit_stacksize_unlimited.err
ulimit_stacksize_unlimited/f2782a7c: performing regular expression - '^unlimited$' on
↳file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳tests/generic.local.bash/ulimits/ulimit_stacksize_unlimited/f2782a7c/ulimit_stacksize_
↳unlimited.out with regular expression
ulimit_stacksize_unlimited/f2782a7c: Regular Expression Match - Failed!
ulimit_max_user_process_2048/7d195a72: completed with returncode: 0
ulimit_max_user_process_2048/7d195a72: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_max_user_process_2048/7d195a72/ulimit_max_user_process_2048.out
ulimit_max_user_process_2048/7d195a72: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_max_user_process_2048/7d195a72/ulimit_max_user_process_2048.err
ulimit_max_user_process_2048/7d195a72: performing regular expression - '^2048$' on file:
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/ulimits/ulimit_max_user_process_2048/7d195a72/ulimit_max_user_
↳process_2048.out with regular expression
ulimit_max_user_process_2048/7d195a72: Regular Expression Match - Failed!
variables_bash/7ff48178: completed with returncode: 0
variables_bash/7ff48178: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↳7ff48178/variables_bash.out
variables_bash/7ff48178: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↳7ff48178/variables_bash.err
ssh_localhost_remotecommand/93086006: completed with returncode: 255
ssh_localhost_remotecommand/93086006: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ssh_localhost/ssh_localhost_remotecommand/93086006/ssh_localhost_remotecommand.out
ssh_localhost_remotecommand/93086006: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ssh_localhost/ssh_localhost_remotecommand/93086006/ssh_localhost_remotecommand.err
ulimit_cputime_unlimited/3914f699: completed with returncode: 0
ulimit_cputime_unlimited/3914f699: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_cputime_unlimited/3914f699/ulimit_cputime_unlimited.out
ulimit_cputime_unlimited/3914f699: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_cputime_unlimited/3914f699/ulimit_cputime_unlimited.err
ulimit_cputime_unlimited/3914f699: performing regular expression - '^unlimited$' on
↳file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳tests/generic.local.bash/ulimits/ulimit_cputime_unlimited/3914f699/ulimit_cputime_
↳unlimited.out with regular expression
ulimit_cputime_unlimited/3914f699: Regular Expression Match - Success!
ulimit_vmsize_unlimited/55aecca: completed with returncode: 0

```

(continues on next page)

(continued from previous page)

```

ulimit_vmsize_unlimited/55aecca: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_vmsize_unlimited/55aecca/ulimit_vmsize_unlimited.out
ulimit_vmsize_unlimited/55aecca: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/ulimits/
↳ulimit_vmsize_unlimited/55aecca/ulimit_vmsize_unlimited.err
ulimit_vmsize_unlimited/55aecca: performing regular expression - '^unlimited$' on file:
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/ulimits/ulimit_vmsize_unlimited/55aecca/ulimit_vmsize_unlimited.
↳out with regular expression
ulimit_vmsize_unlimited/55aecca: Regular Expression Match - Success!
ulimit_filedescriptor_4096/fe1869b1: completed with returncode: 0
ulimit_filedescriptor_4096/fe1869b1: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_filedescriptor_4096/fe1869b1/ulimit_filedescriptor_4096.out
ulimit_filedescriptor_4096/fe1869b1: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ulimits/ulimit_filedescriptor_4096/fe1869b1/ulimit_filedescriptor_4096.err
ulimit_filedescriptor_4096/fe1869b1: performing regular expression - '^4096$' on file: /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/ulimits/ulimit_filedescriptor_4096/fe1869b1/ulimit_filedescriptor_
↳4096.out with regular expression
ulimit_filedescriptor_4096/fe1869b1: Regular Expression Match - Failed!
systemd_default_target/d016e48c: completed with returncode: 1
systemd_default_target/d016e48c: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-
↳default-target/systemd_default_target/d016e48c/systemd_default_target.out
systemd_default_target/d016e48c: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-
↳default-target/systemd_default_target/d016e48c/systemd_default_target.err
root_disk_usage/edf300cd: completed with returncode: 0
root_disk_usage/edf300cd: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳usage/edf300cd/root_disk_usage.out
root_disk_usage/edf300cd: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳usage/edf300cd/root_disk_usage.err

```

```

-----
Launching test: ssh_localhost_remotecommand
Test ID: 93086006-02fa-4f6b-be39-127476b64961
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ssh_localhost/ssh_localhost_remotecommand/93086006/
↳ssh_localhost_remotecommand_build.sh

```

```

-----
Launching test: kernel_swapusage
Test ID: 5ef212a6-ccd2-41aa-8f1a-29f1d2879522
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/kernel_state/kernel_swapusage/5ef212a6/kernel_
↳swapusage_build.sh

```

(continues on next page)

(continued from previous page)

```
Launching test: ulimit_filelock_unlimited
Test ID: ac5c8bd1-e810-4700-a593-d2de24bf6eb3
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_filelock_unlimited/ac5c8bd1/ulimit_
↳filelock_unlimited_build.sh
```

```
Launching test: ulimit_cputime_unlimited
Test ID: 3914f699-b36a-481c-88a4-0636eb4ef6b3
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_cputime_unlimited/3914f699/ulimit_
↳cputime_unlimited_build.sh
```

```
Launching test: ulimit_stacksize_unlimited
Test ID: f2782a7c-f09e-495d-8905-660b9b4d4813
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_stacksize_unlimited/f2782a7c/ulimit_
↳stacksize_unlimited_build.sh
```

```
Launching test: ulimit_vmsize_unlimited
Test ID: 55aecca-115c-4f90-89e0-d0801271d5dc
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_vmsize_unlimited/55aecca/ulimit_
↳vmsize_unlimited_build.sh
```

```
Launching test: ulimit_filedescriptor_4096
Test ID: fe1869b1-aa96-466f-b704-b4b0a1ae40f0
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_filedescriptor_4096/fe1869b1/ulimit_
↳filedescriptor_4096_build.sh
```

```
Launching test: ulimit_max_user_process_2048
Test ID: 7d195a72-59e1-45a9-9302-7e283f601c7b
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ulimits/ulimit_max_user_process_2048/7d195a72/ulimit_
↳max_user_process_2048_build.sh
```

```
Launching test: systemd_default_target
Test ID: d016e48c-20f7-4111-8b5d-db8b8ded804d
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/systemd-default-target/systemd_default_target/
↳d016e48c/systemd_default_target_build.sh
```

```
Launching test: variables_bash
Test ID: 7ff48178-2f82-4dc0-8e9b-17ce6ff0b48f
Executor Name: generic.local.bash
```

(continues on next page)

(continued from previous page)

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash_build.sh

Launching test: root_disk_usage

Test ID: edf300cd-7242-47a3-8924-4150e812a4af

Executor Name: generic.local.bash

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_usage/edf300cd/root_disk_usage_build.sh

+-----+
| Stage: Test Summary |
+-----+

name	id	executor	status	returncode_
match	regex_match	runtime_match	returncode	runtime
ssh_localhost_remotecommand	93086006	generic.local.bash	FAIL	N/A
N/A	N/A	255	0.032854	
kernel_swapusage	5ef212a6	generic.local.bash	FAIL	N/A
N/A	N/A	127	0.031774	
ulimit_filelock_unlimited	ac5c8bd1	generic.local.bash	PASS	False
True	False	0	0.018337	
ulimit_cputime_unlimited	3914f699	generic.local.bash	PASS	False
True	False	0	0.019287	
ulimit_stacksize_unlimited	f2782a7c	generic.local.bash	FAIL	False
False	False	0	0.023499	
ulimit_vmsize_unlimited	55aeecca	generic.local.bash	PASS	False
True	False	0	0.022004	
ulimit_filedescriptor_4096	fe1869b1	generic.local.bash	FAIL	False
False	False	0	0.027482	
ulimit_max_user_process_2048	7d195a72	generic.local.bash	FAIL	False
False	False	0	0.02491	
systemd_default_target	d016e48c	generic.local.bash	FAIL	N/A
N/A	N/A	1	0.030873	
variables_bash	7ff48178	generic.local.bash	PASS	N/A
N/A	N/A	0	0.042796	
root_disk_usage	edf300cd	generic.local.bash	PASS	N/A
N/A	N/A	0	0.02729	

Passed Tests: 5/11 Percentage: 45.455%

Failed Tests: 6/11 Percentage: 54.545%

Writing Logfile to: /tmp/buildtest_hkdfz2sm.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

Excluding Buildspects

So far we learned how to build buildspects by file and directory path using the `-b` option. Next, we will discuss how one may exclude buildspects which behaves similar to `-b` option. You can exclude buildspects via `--exclude` or short option `-x` which can be useful when you want to exclude certain files or sub directory.

For example we can build all buildspects in `tutorials` but exclude file `tutorials/vars.yml` by running:

```
$ buildtest build -b tutorials -x tutorials/vars.yml
```

buildtest will discover all buildspects and then exclude any buildspects specified by `-x` option. You can specify `-x` multiple times just like `-b` option.

For example, we can undo discovery by passing same option to `-b` and `-x` as follows

```
$ buildtest bd -b tutorials/ -x tutorials/
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:14
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳bin/buildtest bd -b tutorials/ -x tutorials/
There are no Buildspec files to process.
```

Buildtest will stop immediately if there are no Buildspects to process, this is true if you were to specify files instead of directory.

In this example, we build all buildspects in a directory but exclude a file. Buildtest will report the excluded buildspects in the output and `-x` option can be appended multiple times. The `-x` can be a file or a directory and behaves similar to `-b` option.

```
$ buildtest bd -b general_tests/configuration/ -x general_tests/configuration/ulimits.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:14
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/buildtest/settings/config.yml
```

(continues on next page)

(continued from previous page)

```
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest bd -b general_tests/configuration/ -x general_tests/configuration/
↳ ulimits.yml
```

```
+-----+
| Stage: Discovering Buildsspecs |
+-----+
```

```
+-----+
↳ -----+
| Discovered Buildsspecs |
```

```
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ulimits.yml |
```

```
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/kernel_state.yml |
```

```
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/disk_usage.yml |
```

```
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/systemd-default-target.yml |
```

```
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ssh_localhost.yml |
```

```
+-----+
↳ -----+
| Excluded Buildsspecs |
```

```
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ulimits.yml |
```

```
+-----+
Discovered Buildsspecs: 5
Excluded Buildsspecs: 1
Detected Buildsspecs after exclusion: 4
```

```
+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

```
Valid Buildsspecs: 4
Invalid Buildsspecs: 0
```

(continues on next page)

(continued from previous page)

```

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/kernel_state.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/disk_usage.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/systemd-default-target.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ssh_localhost.yml: VALID

```

Total builder objects created: 4

```

builders: [kernel_swapusage/92b2bafa, root_disk_usage/1cd41359, systemd_default_target/
↳ e16f7e53, ssh_localhost_remotecommand/c07d8422]

```

name	id	description
↳ buildspeccs		

kernel_swapusage	92b2bafa	Retrieve Kernel Swap Usage
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_tests/configuration/kernel_state.yml		
root_disk_usage	1cd41359	Check root disk usage and report if it exceeds
↳ threshold		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_tests/configuration/disk_usage.yml
systemd_default_target	e16f7e53	check if default target is multi-user.target
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_tests/configuration/systemd-default-target.yml		
ssh_localhost_remotecommand	c07d8422	Test if ssh on localhost works and if we can run
↳ remote command.		/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_tests/configuration/ssh_localhost.yml

```

+-----+
| Stage: Building Test |
+-----+

```

name	id	type	executor	tags
↳ testpath				

kernel_swapusage	92b2bafa	script	generic.local.bash	['configuration', 'kernel_swapusage_build.sh']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/kernel_swapusage/92b2bafa/kernel_swapusage_build.sh				
root_disk_usage	1cd41359	script	generic.local.bash	['filesystem', 'storage', 'root_disk_usage_build.sh']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_usage/1cd41359/root_disk_usage_build.sh				
systemd_default_target	e16f7e53	script	generic.local.bash	['system', 'systemd_default_target_build.sh']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-default-target/systemd_default_target/e16f7e53/systemd_default_target_build.sh				

(continued from previous page)

```

ssh_localhost_remotecommand | c07d8422 | script | generic.local.bash | ['ssh']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳ 0/var/tests/generic.local.bash/ssh_localhost/ssh_localhost_remotecommand/c07d8422/ssh_
↳ localhost_remotecommand_build.sh

+-----+
| Stage: Running Test |
+-----+

kernel_swapusage/92b2bafa: completed with returncode: 127
kernel_swapusage/92b2bafa: Writing output file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/
↳ kernel_swapusage/92b2bafa/kernel_swapusage.out
kernel_swapusage/92b2bafa: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/kernel_state/kernel_
↳ swapusage/92b2bafa/kernel_swapusage.err
systemd_default_target/e16f7e53: completed with returncode: 1
systemd_default_target/e16f7e53: Writing output file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-
↳ default-target/systemd_default_target/e16f7e53/systemd_default_target.out
systemd_default_target/e16f7e53: Writing error file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/systemd-
↳ default-target/systemd_default_target/e16f7e53/systemd_default_target.err
root_disk_usage/1cd41359: completed with returncode: 0
root_disk_usage/1cd41359: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳ usage/1cd41359/root_disk_usage.out
root_disk_usage/1cd41359: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳ usage/1cd41359/root_disk_usage.err
ssh_localhost_remotecommand/c07d8422: completed with returncode: 255
ssh_localhost_remotecommand/c07d8422: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ ssh_localhost/ssh_localhost_remotecommand/c07d8422/ssh_localhost_remotecommand.out
ssh_localhost_remotecommand/c07d8422: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ ssh_localhost/ssh_localhost_remotecommand/c07d8422/ssh_localhost_remotecommand.err

-----
Launching test: kernel_swapusage
Test ID: 92b2bafa-44d0-42c6-84c5-272b1a2ba768
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/kernel_state/kernel_swapusage/92b2bafa/kernel_
↳ swapusage_build.sh

-----
Launching test: root_disk_usage
Test ID: 1cd41359-a069-43ea-8807-2b6f256c34e3
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/disk_usage/root_disk_usage/1cd41359/root_disk_usage_
↳ build.sh

-----

```

(continues on next page)

(continued from previous page)

```

Launching test: systemd_default_target
Test ID: e16f7e53-2ed2-46be-b2e6-6212c0e19fc9
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/systemd-default-target/systemd_default_target/
↳e16f7e53/systemd_default_target_build.sh

```

```

Launching test: ssh_localhost_remotecommand
Test ID: c07d8422-61be-4bdd-893b-10387062609c
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/ssh_localhost/ssh_localhost_remotecommand/c07d8422/
↳ssh_localhost_remotecommand_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_
↳match	↳regex_match	↳runtime_match	↳returncode	↳runtime
kernel_swapusage	92b2bafa	generic.local.bash	FAIL	N/A
↳	N/A	127	0.027239	
root_disk_usage	1cd41359	generic.local.bash	PASS	N/A
↳	N/A	0	0.037134	
systemd_default_target	e16f7e53	generic.local.bash	FAIL	N/A
↳	N/A	1	0.027696	
ssh_localhost_remotecommand	c07d8422	generic.local.bash	FAIL	N/A
↳	N/A	255	0.030714	

```

Passed Tests: 1/4 Percentage: 25.000%
Failed Tests: 3/4 Percentage: 75.000%

```

```

Writing Logfile to: /tmp/buildtest_cbtfxlm.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

```

Building By Tags

buildtest can perform builds by tags by using `--tags` or short option `(-t)`. In order to use this feature, buildtest must load buildsspecs in *cache* which can be run via `buildtest buildspec find`. If you are unsure of the available tags you can run `buildtest buildspec find --tags` or let buildtest tab-complete the available tags. For more details see *Querying buildspec tags*.

Let's assume you want to build by tag name `network`, buildtest will attempt to find all tests that contain tags: `['network']` in the buildspec which is loaded in the buildcache cache. If a test matches the tag name, the test will be picked up during the discover process.

```

$ buildtest build -t network
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:14
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -t network

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ tags_example.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

BREAKDOWN OF BUILDSPESCS BY TAGS
-----
Detected Tag Names: ['network']
+-----+
↳ -----+
| network |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ tags_example.yml |
+-----+
↳ -----+

+-----+
| Stage: Parsing Buildsspecs |
+-----+

```

(continues on next page)

(continued from previous page)

```
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ tags_example.yml: VALID
```

```
Total builder objects created: 2
builders: [string_tag/3d7a4f9f, list_of_strings_tags/fc9ecab8]
```

name	id	description	buildspecs
string_tag	3d7a4f9f	tags can be a string	/home/docs/checkouts/ ↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/tags_example.yml
list_of_strings_tags	fc9ecab8	tags can be a list of strings	/home/docs/checkouts/ ↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/tags_example.yml

```
+-----+
| Stage: Building Test |
+-----+
```

name	id	type	executor	tags	↳
testpath					
string_tag	3d7a4f9f	script	generic.local.bash	network	/
list_of_strings_tags	fc9ecab8	script	generic.local.bash	['network', 'ping']	/

```
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳ generic.local.bash/tags_example/string_tag/3d7a4f9f/string_tag_build.sh
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳ generic.local.bash/tags_example/list_of_strings_tags/fc9ecab8/list_of_strings_tags_
↳ build.sh
```

```
+-----+
| Stage: Running Test |
+-----+
```

```
string_tag/3d7a4f9f: completed with returncode: 0
string_tag/3d7a4f9f: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/tags_example/string_
↳ tag/3d7a4f9f/string_tag.out
string_tag/3d7a4f9f: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/tags_example/string_
↳ tag/3d7a4f9f/string_tag.err
list_of_strings_tags/fc9ecab8: completed with returncode: 0
list_of_strings_tags/fc9ecab8: Writing output file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/tags_example/list_
↳ of_strings_tags/fc9ecab8/list_of_strings_tags.out
```

(continues on next page)

(continued from previous page)

```
list_of_strings_tags/fc9ecab8: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/tags_example/list_
↳of_strings_tags/fc9ecab8/list_of_strings_tags.err
```

Launching test: string_tag

Test ID: 3d7a4f9f-a80c-4ce5-b762-679d5dec1149

Executor Name: generic.local.bash

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/tags_example/string_tag/3d7a4f9f/string_tag_build.sh

Launching test: list_of_strings_tags

Test ID: fc9ecab8-f728-4a44-8b1f-1c116d67c2cb

Executor Name: generic.local.bash

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/tags_example/list_of_strings_tags/fc9ecab8/list_of_
↳strings_tags_build.sh

```
+-----+
| Stage: Test Summary |
+-----+
```

name	id	executor	status	returncode_match	
↳regex_match	↳runtime_match	↳returncode	↳runtime		
string_tag	3d7a4f9f	generic.local.bash	PASS	N/A	↳
↳N/A	N/A	0	0.032556		
list_of_strings_tags	fc9ecab8	generic.local.bash	PASS	N/A	↳
↳N/A	N/A	0	3.06947		

Passed Tests: 2/2 Percentage: 100.000%

Failed Tests: 0/2 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_3e2c2fhc.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

You can build by multiple tags by specifying --tags multiple times. In next example we build all tests with tag name pass and python.

```
$ buildtest build -t python -t pass
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:18
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳python
```

(continues on next page)

(continued from previous page)

```

python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳bin/buildtest build -t python -t pass

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳-----+
| Discovered Buildsspecs |
↳
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml |
+-----+
↳-----+
Discovered Buildsspecs: 3
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 3

BREAKDOWN OF BUILDSPECS BY TAGS
-----
Detected Tag Names: ['python', 'pass']
+-----+
↳-----+
| python |
↳
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml |
+-----+
↳-----+
+-----+
↳-----+

```

(continues on next page)

(continued from previous page)

```

| pass
↪
=====
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml |
+-----+
↪-----+

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 3
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-shell.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml: VALID

Total builder objects created: 6
builders: [circle_area/6ed8bf63, python_hello/6feaf262, exit1_fail/aaa388f4, exit1_pass/
↪55601e7e, returncode_list_mismatch/5e2d005d, returncode_int_match/635d49bb]

name                id                description
↪buildspecs
-----
↪-----
circle_area          6ed8bf63  Calculate circle of area given a radius
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-shell.yml
python_hello         6feaf262  Hello World python
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-hello.yml
exit1_fail           aaa388f4  exit 1 by default is FAIL
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml
exit1_pass           55601e7e  report exit 1 as PASS
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml
returncode_list_mismatch 5e2d005d  exit 2 failed since it failed to match returncode 1
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml
returncode_int_match 635d49bb  exit 128 matches returncode 128
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml

```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags          |
+-----+-----+-----+-----+-----+
| testpath
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
circle_area    | 6ed8bf63 | script | generic.local.python | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.bash/python-hello/python_hello/6feaf262/python_hello_build.sh
exit1_fail     | aaa388f4 | script | generic.local.sh     | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.bash/python-hello/python_hello/6feaf262/python_hello_build.sh
exit1_pass     | 55601e7e | script | generic.local.sh     | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.sh/pass_returncode/exit1_fail/aaa388f4/exit1_fail_build.sh
exit1_pass     | 55601e7e | script | generic.local.sh     | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.sh/pass_returncode/exit1_pass/55601e7e/exit1_pass_build.sh
returncode_list_mismatch | 5e2d005d | script | generic.local.sh     | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.sh/pass_returncode/exit1_fail/aaa388f4/exit1_fail_build.sh
returncode_list_mismatch | 5e2d005d | script | generic.local.sh     | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/5e2d005d/
returncode_list_mismatch_build.sh
returncode_int_match    | 635d49bb | script | generic.local.sh     | ['tutorials',
python_hello   | 6feaf262 | script | generic.local.bash   | python
+-----+-----+-----+-----+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
var/tests/generic.local.sh/pass_returncode/returncode_int_match/635d49bb/returncode_
returncode_int_match_build.sh

+-----+
| Stage: Running Test |
+-----+

python_hello/6feaf262: completed with returncode: 0
python_hello/6feaf262: Writing output file: /home/docs/checkouts/readthedocs.org/user_
builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
hello/6feaf262/python_hello.out
python_hello/6feaf262: Writing error file: /home/docs/checkouts/readthedocs.org/user_
builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
hello/6feaf262/python_hello.err
exit1_fail/aaa388f4: completed with returncode: 1
exit1_fail/aaa388f4: Writing output file: /home/docs/checkouts/readthedocs.org/user_
builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
fail/aaa388f4/exit1_fail.out
exit1_fail/aaa388f4: Writing error file: /home/docs/checkouts/readthedocs.org/user_
builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
fail/aaa388f4/exit1_fail.err
returncode_list_mismatch/5e2d005d: completed with returncode: 2
returncode_list_mismatch/5e2d005d: Writing output file: /home/docs/checkouts/readthedocs.
org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
returncode_list_mismatch/5e2d005d/returncode_list_mismatch.out

```

(continued from previous page)

```

returncode_list_mismatch/5e2d005d: Writing error file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/5e2d005d/returncode_list_mismatch.err
returncode_list_mismatch/5e2d005d: Checking returncode - 2 is matched in list [1, 3]
circle_area/6ed8bf63: completed with returncode: 0
circle_area/6ed8bf63: Writing output file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_area/6ed8bf63/circle_area.out
circle_area/6ed8bf63: Writing error file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_area/6ed8bf63/circle_area.err
exit1_pass/55601e7e: completed with returncode: 1
exit1_pass/55601e7e: Writing output file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/55601e7e/exit1_pass.out
exit1_pass/55601e7e: Writing error file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/55601e7e/exit1_pass.err
exit1_pass/55601e7e: Checking returncode - 1 is matched in list [1]
returncode_int_match/635d49bb: completed with returncode: 128
returncode_int_match/635d49bb: Writing output file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/635d49bb/returncode_int_match.out
returncode_int_match/635d49bb: Writing error file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/635d49bb/returncode_int_match.err
returncode_int_match/635d49bb: Checking returncode - 128 is matched in list [128]

-----
Launching test: circle_area
Test ID: 6ed8bf63-8008-42c7-9416-7e472c15c9e4
Executor Name: generic.local.python
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_area/6ed8bf63/circle_area_build.sh

-----
Launching test: python_hello
Test ID: 6feaf262-d168-4637-ba18-0646a62ab5e1
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_hello/6feaf262/python_hello_build.sh

-----
Launching test: exit1_fail
Test ID: aaa388f4-6e71-4e73-aabd-4f5fb8e317ad
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_fail/aaa388f4/exit1_fail_build.sh

-----
Launching test: exit1_pass
Test ID: 55601e7e-a451-4fc0-81bb-5138115297a9
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/55601e7e/exit1_pass_build.sh

```

(continued from previous page)

```

Launching test: returncode_list_mismatch
Test ID: 5e2d005d-b672-4298-a6d4-245346eed5d5
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/5e2d005d/
↳returncode_list_mismatch_build.sh

```

```

Launching test: returncode_int_match
Test ID: 635d49bb-bd3c-4e94-bd1d-87080a25713a
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/635d49bb/
↳returncode_int_match_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_
↳match	↳regex_match	↳runtime_match	↳returncode	↳runtime
python_hello	6feaf262	generic.local.bash	PASS	N/A
↳ N/A	↳ N/A	↳ 0	↳ 0.102769	
exit1_fail	aaa388f4	generic.local.sh	FAIL	N/A
↳ N/A	↳ N/A	↳ 1	↳ 0.029501	
exit1_pass	55601e7e	generic.local.sh	PASS	True
↳ False	↳ False	↳ 1	↳ 0.018127	
returncode_list_mismatch	5e2d005d	generic.local.sh	FAIL	False
↳ False	↳ False	↳ 2	↳ 0.018428	
returncode_int_match	635d49bb	generic.local.sh	PASS	True
↳ False	↳ False	↳ 128	↳ 0.023221	
circle_area	6ed8bf63	generic.local.python	PASS	N/A
↳ N/A	↳ N/A	↳ 0	↳ 0.116362	

Passed Tests: 4/6 Percentage: 66.667%

Failed Tests: 2/6 Percentage: 33.333%

Writing Logfile to: /tmp/buildtest_7pfzvc1_.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

When multiple tags are specified, we search each tag independently and if it's found in the builds spec cache we retrieve the builds spec file and add file to queue. This queue is a list of builds specs that buildtest will process (i.e parse, build, run). You can *query tags* from builds specs cache to see all available tags by running `buildtest builds spec find --tags`.

Note: The `--tags` is used for discovering buildspec file and not filtering tests by tag.

You can combine `--tags` with `--buildspec` to discover buildspecs in a single command. buildtest will query tags and buildspecs independently and combine all discovered buildspecs together.

```
$ buildtest build --tags pass --buildspec tutorials/python-hello.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:18
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build --tags pass --buildspec tutorials/python-hello.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-hello.yml |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 2
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 2

BREAKDOWN OF BUILDSPECS BY TAGS
-----
Detected Tag Names: ['pass']
+-----+
↳ -----+
| pass |
↳ |
+=====+
```

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml |
+-----+
↳-----+

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 2
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml: VALID

Total builder objects created: 5
builders: [python_hello/3c422a2b, exit1_fail/46c30f6b, exit1_pass/b2c32941, returncode_
↳list_mismatch/1f4bb7da, returncode_int_match/1cc26cef]

name          id          description
↳buildspecs
-----
↳-----
python_hello    3c422a2b  Hello World python
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml
exit1_fail      46c30f6b  exit 1 by default is FAIL
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml
exit1_pass      b2c32941  report exit 1 as PASS
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml
returncode_list_mismatch 1f4bb7da  exit 2 failed since it failed to match returncode 1
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml
returncode_int_match 1cc26cef  exit 128 matches returncode 128
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type  | executor          | tags
↳ | testpath
-----+-----+-----+-----+-----
↳-----
```

(continues on next page)

(continued from previous page)

```

python_hello          | 3c422a2b | script | generic.local.bash | python
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.bash/python-hello/python_hello/3c422a2b/python_hello_build.sh
exit1_fail            | 46c30f6b | script | generic.local.sh   | ['tutorials', 'fail
↳ '] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/exit1_fail/46c30f6b/exit1_fail_build.sh
exit1_pass            | b2c32941 | script | generic.local.sh   | ['tutorials', 'pass
↳ '] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/exit1_pass/b2c32941/exit1_pass_build.sh
returncode_list_mismatch | 1f4bb7da | script | generic.local.sh   | ['tutorials', 'fail
↳ '] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/returncode_list_mismatch/1f4bb7da/returncode_
↳ list_mismatch_build.sh
returncode_int_match   | 1cc26cef | script | generic.local.sh   | ['tutorials', 'pass
↳ '] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/returncode_int_match/1cc26cef/returncode_int_
↳ match_build.sh

+-----+
| Stage: Running Test |
+-----+

exit1_fail/46c30f6b: completed with returncode: 1
exit1_fail/46c30f6b: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ fail/46c30f6b/exit1_fail.out
exit1_fail/46c30f6b: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ fail/46c30f6b/exit1_fail.err
exit1_pass/b2c32941: completed with returncode: 1
exit1_pass/b2c32941: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/b2c32941/exit1_pass.out
exit1_pass/b2c32941: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/b2c32941/exit1_pass.err
exit1_pass/b2c32941: Checking returncode - 1 is matched in list [1]
returncode_list_mismatch/1f4bb7da: completed with returncode: 2
returncode_list_mismatch/1f4bb7da: Writing output file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/1f4bb7da/returncode_list_mismatch.out
returncode_list_mismatch/1f4bb7da: Writing error file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/1f4bb7da/returncode_list_mismatch.err
returncode_list_mismatch/1f4bb7da: Checking returncode - 2 is matched in list [1, 3]
python_hello/3c422a2b: completed with returncode: 0
python_hello/3c422a2b: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
↳ hello/3c422a2b/python_hello.out
python_hello/3c422a2b: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
↳ hello/3c422a2b/python_hello.err

```

(continues on next page)

(continued from previous page)

```
returncode_int_match/1cc26cef: completed with returncode: 128
returncode_int_match/1cc26cef: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_int_match/1cc26cef/returncode_int_match.out
returncode_int_match/1cc26cef: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_int_match/1cc26cef/returncode_int_match.err
returncode_int_match/1cc26cef: Checking returncode - 128 is matched in list [128]
```

```
-----
Launching test: python_hello
Test ID: 3c422a2b-81b4-4d4f-9240-1e74655d9c8e
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/python_hello_
↳build.sh
```

```
-----
Launching test: exit1_fail
Test ID: 46c30f6b-5121-4715-90b8-961091a04665
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/exit1_fail/46c30f6b/exit1_fail_build.sh
```

```
-----
Launching test: exit1_pass
Test ID: b2c32941-222c-40cc-ada7-8e034ae8b5e9
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/b2c32941/exit1_pass_build.sh
```

```
-----
Launching test: returncode_list_mismatch
Test ID: 1f4bb7da-00c0-4154-9951-95a94ff144f4
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/1f4bb7da/
↳returncode_list_mismatch_build.sh
```

```
-----
Launching test: returncode_int_match
Test ID: 1cc26cef-e085-42ff-a182-bdd075dce883
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/1cc26cef/
↳returncode_int_match_build.sh
```

```
+-----+
| Stage: Test Summary |
+-----+
```

name	id	executor	status	returncode_match
↳ regex_match	runtime_match	returncode	runtime	

↳ +-----+				
python_hello	3c422a2b	generic.local.bash	PASS	N/A
↳ N/A	N/A	0	0.092008	

(continues on next page)

(continued from previous page)

exit1_fail	46c30f6b	generic.local.sh	FAIL	N/A	↵
↵ N/A	N/A		1 0.037522		
exit1_pass	b2c32941	generic.local.sh	PASS	True	↵
↵ False	False		1 0.034106		
returncode_list_mismatch	1f4bb7da	generic.local.sh	FAIL	False	↵
↵ False	False		2 0.017392		
returncode_int_match	1cc26cef	generic.local.sh	PASS	True	↵
↵ False	False		128 0.013467		

Passed Tests: 3/5 Percentage: 60.000%

Failed Tests: 2/5 Percentage: 40.000%

Writing Logfile to: /tmp/buildtest_tg3mt2bz.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↵readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

As you may see, there are several ways to build buildsspecs with buildtest. Tags is great way to build a whole collection of tests if you don't know path to all the files. You can specify multiple tags per buildsspecs to classify how test can be run.

Building by Executors

Every buildspec is associated to an executor which is responsible for running the test. You can instruct buildtest to run all tests by given executor via `--executor` option or short option `-e`. For instance, if you want to build all test associated to executor `generic.local.python` you can run:

```
$ buildtest build --executor generic.local.python
```

buildtest will query buildspec cache for the executor name and retrieve a list of buildsspecs with matching executor name. To see a list of available executors in buildspec cache see [querying buildspec executor](#).

Note: By default all tests are run in buildspec file. The `buildtest build --executor` option discovers buildsspecs if one of the test matches the executor name. The `--executor` option is **not filtering tests but only discovering buildsspecs**.

In this example we run all tests that are associated to `generic.local.python` executor.

```
$ buildtest build --executor generic.local.python
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:19
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↵11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↵python
python version: 3.7.9
```

(continues on next page)

(continued from previous page)

```

Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build --executor generic.local.python

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-shell.yml |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_platform.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 2
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 2

BREAKDOWN OF BUILDSPECS BY EXECUTORS
-----
Detected Executor Names: ['generic.local.python']
+-----+
↳ -----+
| generic.local.python |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-shell.yml |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_platform.yml |
+-----+
↳ -----+

+-----+
| Stage: Parsing Buildsspecs |
+-----+

[run_only_platform_darwin][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/run_only_platform.yml]: test is skipped because this test
↳ is expected to run on platform: Darwin but detected platform: Linux.

```

(continued from previous page)

```

Valid Buildsspecs: 2
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳run_only_platform.yml: VALID

Total builder objects created: 2
builders: [circle_area/29fd70b3, run_only_platform_linux/33c3523e]

name          id          description
↳buildspecs
-----
↳-----
↳-----
circle_area    29fd70b3  Calculate circle of area given a radius /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml
run_only_platform_linux  33c3523e  This test will only run if target platform is Linux /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳run_only_platform.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type  | executor          | tags
↳          | testpath
-----+-----+-----+-----+-----
↳+-----+-----+-----+-----+-----
↳-----
↳-----
circle_area    | 29fd70b3 | script | generic.local.python | ['tutorials',
↳'python'] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/var/tests/generic.local.python/python-shell/circle_area/29fd70b3/circle_area_build.sh
run_only_platform_linux | 33c3523e | script | generic.local.python | ['tutorials']
↳          | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳var/tests/generic.local.python/run_only_platform/run_only_platform_linux/33c3523e/run_
↳only_platform_linux_build.sh

+-----+
| Stage: Running Test |
+-----+

circle_area/29fd70b3: completed with returncode: 0
circle_area/29fd70b3: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/29fd70b3/circle_area.out
circle_area/29fd70b3: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/29fd70b3/circle_area.err

```

(continues on next page)

(continued from previous page)

```

run_only_platform_linux/33c3523e: completed with returncode: 0
run_only_platform_linux/33c3523e: Writing output file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/run_only_
↳platform/run_only_platform_linux/33c3523e/run_only_platform_linux.out
run_only_platform_linux/33c3523e: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/run_only_
↳platform/run_only_platform_linux/33c3523e/run_only_platform_linux.err
run_only_platform_linux/33c3523e: performing regular expression - '^Linux' on file: /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.python/run_only_platform/run_only_platform_linux/33c3523e/run_only_
↳platform_linux.out with regular expression
run_only_platform_linux/33c3523e: Regular Expression Match - Success!

```

```

-----
Launching test: circle_area

```

```

Test ID: 29fd70b3-2bd5-45a0-96af-0b62e394c87e

```

```

Executor Name: generic.local.python

```

```

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.python/python-shell/circle_area/29fd70b3/circle_area_
↳build.sh

```

```

-----
Launching test: run_only_platform_linux

```

```

Test ID: 33c3523e-e52b-4bad-a118-c048b0ac5d88

```

```

Executor Name: generic.local.python

```

```

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.python/run_only_platform/run_only_platform_linux/33c3523e/
↳run_only_platform_linux_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_match_
↳ regex_match	↳ runtime_match	↳ returncode	↳ runtime	
run_only_platform_linux	33c3523e	generic.local.python	PASS	False
↳ True	↳ False	↳ 0	↳ 0.141286	
circle_area	29fd70b3	generic.local.python	PASS	N/A
↳ N/A	↳ N/A	↳ 0	↳ 0.115076	

```

Passed Tests: 2/2 Percentage: 100.000%

```

```

Failed Tests: 0/2 Percentage: 0.000%

```

```

Writing Logfile to: /tmp/buildtest_f5kdyyj_.log

```

```

A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

```

Note: The --executor option can be appended to discover tests by multiple executors.

Filtering Buildsspecs

buildtest has support for filtering buildsspecs based on certain attributes defined in buildspec file. Upon *Discover Buildsspecs*, buildtest will filter out tests or entire buildspec files. The `buildtest build --filter` option can be used to filter buildsspecs which expects a **single** key=value pair. Currently, buildtest can filter tests based on `tags`, `type` and `maintainers`.

To see all available filter fields you can run `buildtest build --helpfilter` and buildtest will report the fields followed by description.

```
$ buildtest build --helpfilter
Field      Description
-----
tags       Filter tests by 'tag' field
type       Filter test by 'type' field
maintainers Filter test by 'maintainers' field
```

In this example, we will discover all buildsspecs based on tagname `pass` and then filter each **test** by tagname `pass` specified by `--filter tags=pass`.

```
$ buildtest build -t pass --filter tags=pass
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:19
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -t pass --filter tags=pass

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ | Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml |
+-----+
↳ |
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1
```

(continues on next page)

(continued from previous page)

BREAKDOWN OF BUILDSPECS BY TAGS

Detected Tag Names: ['pass']

```
+-----+
| pass
|
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| pass_returncode.yml |
+-----+
```

```
+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

```
[exit1_fail][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
0/tutorials/pass_returncode.yml]: test is skipped because it is not in tag filter
list: {'tags': 'pass'}
[returncode_list_mismatch][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
checkouts/v0.11.0/tutorials/pass_returncode.yml]: test is skipped because it is not in
tag filter list: {'tags': 'pass'}
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
pass_returncode.yml: VALID
```

Total builder objects created: 2
builders: [exit1_pass/80376cc8, returncode_int_match/f1095054]

name	id	description	buildspecs
exit1_pass	80376cc8	report exit 1 as PASS	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml
returncode_int_match	f1095054	exit 128 matches returncode 128	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml

```
+-----+
| Stage: Building Test |
+-----+
```

name	id	type	executor	tags
testpath				

(continues on next page)

(continued from previous page)

```

exit1_pass          | 80376cc8 | script | generic.local.sh | ['tutorials', 'pass'] | /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.sh/pass_returncode/exit1_pass/80376cc8/exit1_pass_build.sh
returncode_int_match | f1095054 | script | generic.local.sh | ['tutorials', 'pass'] | /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.sh/pass_returncode/returncode_int_match/f1095054/returncode_int_match_
↳build.sh

+-----+
| Stage: Running Test |
+-----+

returncode_int_match/f1095054: completed with returncode: 128
returncode_int_match/f1095054: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_int_match/f1095054/returncode_int_match.out
returncode_int_match/f1095054: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_int_match/f1095054/returncode_int_match.err
returncode_int_match/f1095054: Checking returncode - 128 is matched in list [128]
exit1_pass/80376cc8: completed with returncode: 1
exit1_pass/80376cc8: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳pass/80376cc8/exit1_pass.out
exit1_pass/80376cc8: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳pass/80376cc8/exit1_pass.err
exit1_pass/80376cc8: Checking returncode - 1 is matched in list [1]

-----
Launching test: exit1_pass
Test ID: 80376cc8-2c1c-4946-b76b-e0515c46528e
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/80376cc8/exit1_pass_build.sh

-----
Launching test: returncode_int_match
Test ID: f1095054-1457-49fc-b85e-81156b762f35
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/f1095054/
↳returncode_int_match_build.sh

+-----+
| Stage: Test Summary |
+-----+

name          | id          | executor      | status | returncode_match |
↳regex_match  | runtime_match | returncode    | runtime
-----+-----+-----+-----+-----+
↳-----+-----+-----+-----+-----+
exit1_pass     | 80376cc8    | generic.local.sh | PASS   | True              |
↳False        | False       |                | 1      | 0.021258          |

```

(continues on next page)

(continued from previous page)

```

returncode_int_match | f1095054 | generic.local.sh | PASS      | True      |
↪False              | False          |           128 | 0.021259

Passed Tests: 2/2 Percentage: 100.000%
Failed Tests: 0/2 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_ygzwdshx.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

```

buildtest can run filter tests by *maintainers*, this can be useful if you want to run tests that you are maintainer. The *maintainers* field is set per builds spec and not each test. You can filter maintainers via `--filter maintainers=<MAINTAINER_NAME>`. If the *maintainers* field is not specified the builds spec will be filtered out if `--filter maintainers` is specified. In this next example, we will build all tests for maintainer `@shahzebsiddiqui`.

```

$ buildtest build -b tutorials --filter maintainers=@shahzebsiddiqui
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:20
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↪python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪bin/buildtest build -b tutorials --filter maintainers=@shahzebsiddiqui

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪-----+
| Discovered Buildsspecs |
↪
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪spack/spack_test_specs.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-shell.yml |
+-----+
↪-----+

```

(continues on next page)

(continued from previous page)

```

| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/compiler_exclude.yml          |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/gnu_hello_c.yml              |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_executor.yml                    |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/remove_environment_example.yml    |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/metrics_openmp.yml           |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_buildspec_section.yml           |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳hello_world.yml                         |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/mirror_example.yml                |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳root_user.yml                           |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳environment.yml                         |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml                        |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳run_only_distro.yml                     |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/openmp_hello.yml              |
+-----+
↳-----+

```

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/env_create_manifest.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_tags.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/pre_post_cmds.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳csh_shell_examples.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳run_only_platform.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳executor_regex_script.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳sleep.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/concretize_m4.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳pass_returncode.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳add_numbers.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳tags_example.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/vecadd.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/env_create_directory.yml |
+-----+
↳-----+
```

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/custom_run.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/pre_post_build_run.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_variable.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_test.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳burstbuffer_datawarp_executors.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/gnu_hello_fortran.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_multiple_executor_sbatch.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_regex.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/envvar_override.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/executor_scheduler.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/status_by_executors.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳shell_examples.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/install_zlib.yml |
+-----+
↳-----+
```

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/env_install.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳skip_tests.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/compiler_status_regex.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳shebang.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳runtime_status_test.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_sbatch.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳maintainers_example.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳selinux.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳vars.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳status_regex.yml |
+-----+
↳-----+
Discovered Buildsspecs: 52
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 52

+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

(continues on next page)

(continued from previous page)

```

Valid Buildsspecs: 47
Invalid Buildsspecs: 5
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_test_specs.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-shell.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/compiler_exclude.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/gnu_hello_c.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/remove_environment_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/metrics_openmp.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ hello_world.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/mirror_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ root_user.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ environment.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_distro.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/openmp_hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_create_manifest.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/pre_post_cmds.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ csh_shell_examples.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_platform.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ executor_regex_script.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ sleep.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/concretize_m4.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ add_numbers.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ tags_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/vecadd.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_create_directory.yml: VALID

```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/custom_run.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/pre_post_build_run.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_variable.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_test.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/gnu_hello_fortran.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_multiple_executor_sbatch.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_regex.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/envvar_override.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/executor_scheduler.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/status_by_executors.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳shell_examples.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/install_zlib.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳skip_tests.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/compiler_status_regex.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳shebang.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳runtime_status_test.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_sbatch.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳maintainers_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳selinux.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳vars.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳status_regex.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_executor.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_buildspec_section.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_tags.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳burstbuffer_datawarp_executors.yml: INVALID
```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_install.yml: INVALID
```

Buildspecs that were filtered out

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_test_specs.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-shell.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/compiler_exclude.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/gnu_hello_c.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/remove_environment_example.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/metrics_openmp.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/mirror_example.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ root_user.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ environment.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-hello.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_distro.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/openmp_hello.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_create_manifest.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/pre_post_cmds.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ csh_shell_examples.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_platform.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ executor_regex_script.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ sleep.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/concretize_m4.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ add_numbers.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ tags_example.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/vecadd.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_create_directory.yml
```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/custom_run.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/pre_post_build_run.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ metrics_variable.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_test.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/gnu_hello_fortran.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_multiple_executor_sbatch.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ metrics_regex.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/envvar_override.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/executor_scheduler.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/status_by_executors.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ shell_examples.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/install_zlib.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ skip_tests.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/compiler_status_regex.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ shebang.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ runtime_status_test.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_sbatch.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/multiple_executors.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ maintainers_example.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ selinux.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ status_regex.yml
```

```
Total builder objects created: 1
builders: [hello_world/7c3d7f17]
```

name	id	description	buildspecs
-----	-----	-----	-----

↳ ----- (continues on next page)

(continued from previous page)

```

hello_world 7c3d7f17 hello world example /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/tutorials/hello_world.yml

+-----+
| Stage: Building Test |
+-----+

name      | id      | type  | executor      | tags      | testpath
+-----+-----+-----+-----+-----+-----+
↳ -----
↳ -----
hello_world | 7c3d7f17 | script | generic.local.bash | tutorials | /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ hello_world/hello_world/7c3d7f17/hello_world_build.sh

+-----+
| Stage: Running Test |
+-----+

hello_world/7c3d7f17: completed with returncode: 0
hello_world/7c3d7f17: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/hello_world/hello_
↳ world/7c3d7f17/hello_world.out
hello_world/7c3d7f17: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/hello_world/hello_
↳ world/7c3d7f17/hello_world.err

-----
Launching test: hello_world
Test ID: 7c3d7f17-7dff-4da0-8f72-509cb7b69185
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/hello_world/hello_world/7c3d7f17/hello_world_build.sh

+-----+
| Stage: Test Summary |
+-----+

name      | id      | executor      | status  | returncode_match | regex_
↳ match   | runtime_match | returncode | runtime
+-----+-----+-----+-----+-----+-----+
↳ -----
hello_world | 7c3d7f17 | generic.local.bash | PASS    | N/A              | N/A
↳      | N/A      |      0 | 0.006251
↳

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_pdw2691g.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log (continues on next page)

```

(continued from previous page)

Please see [Query Maintainers](#) on list of maintainers and breakdown of buildsspecs by maintainers.

We can also filter tests by `type` field in the builds spec which corresponds to the schema type. In this next example, we filter all tests by spack schema type by passing option `--filter type=spack`. We inform buildtest to stop after build stage (`--stage=build`) for more details see [Configure Build Stages](#).

```
$ buildtest build -b tutorials --filter type=spack --stage=build
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:20
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials --filter type=spack --stage=build

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ +-----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ metrics_regex.yml |
+-----+
↳ +-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/envvar_override.yml |
+-----+
↳ +-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/compiler_exclude.yml |
+-----+
↳ +-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ invalid_executor.yml |
+-----+
↳ +-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_test_specs.yml |
```

(continues on next page)

(continued from previous page)

```
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/pre_post_cmds.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ executor_regex_script.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ runtime_status_test.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ metrics_variable.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ burstbuffer_datawarp_executors.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/pre_post_build_run.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ csh_shell_examples.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ root_user.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/gnu_hello_fortran.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-hello.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ invalid_tags.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/custom_run.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ hello_world.yml |
```

(continues on next page)

(continued from previous page)

```
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳environment.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/metrics_openmp.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳shell_examples.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/openmp_hello.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳skip_tests.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳run_only_distro.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/gnu_hello_c.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳tags_example.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/remove_environment_example.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/concretize_m4.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/install_zlib.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/spack_sbatch.yml |
```

(continues on next page)

(continued from previous page)

```
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪spack/spack_test.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪shebang.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-shell.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪selinux.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪spack/spack_multiple_executor_sbatch.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪invalid_buildspec_section.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪maintainers_example.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪spack/env_install.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪sleep.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪status_regex.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪run_only_platform.yml |
+-----+
↪-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪add_numbers.yml |
```

(continues on next page)

(continued from previous page)

```
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳vars.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/env_create_directory.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/status_by_executors.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/vecadd.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/mirror_example.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/env_create_manifest.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/executor_scheduler.yml |
+-----+
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/compiler_status_regex.yml |
+-----+
+-----+
Discovered Buildsspecs: 52
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 52

+-----+
| Stage: Parsing Buildsspecs |
+-----+

[metric_regex_example][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/metrics_regex.yml]: test is skipped because it is not in
↳type filter list: spack
[override_environmentvars][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compilers/envvar_override.yml]: test is skipped because it
↳is not in type filter list: spack
[vecadd_gnu_exclude][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compilers/compiler_exclude.yml]: test is skipped because
↳it is not in type filter list: spack
[executor_regex_script_schema][/home/docs/checkouts/readthedocs.org/user_builds/
↳buildtest/checkouts/v0.11.0/tutorials/executor_regex_script.yml]: test is skipped
↳because it is not in type filter list: spack
```

(continued from previous page)

```
[timelimit_min_max][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/tutorials/runtime_status_test.yml]: test is skipped because it is not in type.
↳filter list: spack
[timelimit_min][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/tutorials/runtime_status_test.yml]: test is skipped because it is not in type.
↳filter list: spack
[timelimit_max][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/tutorials/runtime_status_test.yml]: test is skipped because it is not in type.
↳filter list: spack
[timelimit_min_fail][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml]: test is skipped because it is
↳not in type filter list: spack
[timelimit_max_fail][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml]: test is skipped because it is
↳not in type filter list: spack
[metric_variable_assignment][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/metrics_variable.yml]: test is skipped because it is not
↳in type filter list: spack
[pre_post_build_run][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compiler/pre_post_build_run.yml]: test is skipped because
↳it is not in type filter list: spack
[csh_shell][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/csh_shell_examples.yml]: test is skipped because it is not in type filter
↳list: spack
[run_only_as_root][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/tutorials/root_user.yml]: test is skipped because it is not in type filter
↳list: spack
[hello_f][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/compiler/gnu_hello_fortran.yml]: test is skipped because it is not in type
↳filter list: spack
[python_hello][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/tutorials/python_hello.yml]: test is skipped because it is not in type filter
↳list: spack
[custom_run_by_compilers][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compiler/custom_run.yml]: test is skipped because it is
↳not in type filter list: spack
[hello_world][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/hello_world.yml]: test is skipped because it is not in type filter list:
↳spack
[bash_env_variables][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/environment.yml]: test is skipped because it is not in
↳type filter list: spack
[csh_env_declaration][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/environment.yml]: test is skipped because it is not in
↳type filter list: spack
[tcsh_env_declaration][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/environment.yml]: test is skipped because it is not in
↳type filter list: spack
[metrics_variable_compiler][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compiler/metrics_openmp.yml]: test is skipped because it
↳is not in type filter list: spack
[_bin_sh_shell][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/tutorials/shell_examples.yml]: test is skipped because it is not in type filter
↳list: spack
```

(continued on next page)

(continued from previous page)

```
[_bin_bash_shell][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/tutorials/shell_examples.yml]: test is skipped because it is not in type
↳filter list: spack
[bash_shell][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/shell_examples.yml]: test is skipped because it is not in type filter
↳list: spack
[sh_shell][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/shell_examples.yml]: test is skipped because it is not in type filter list:
↳spack
[shell_options][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/tutorials/shell_examples.yml]: test is skipped because it is not in type filter
↳list: spack
[openmp_hello_c_example][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compilers/openmp_hello.yml]: test is skipped because it is
↳not in type filter list: spack
[skip](/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/skip_tests.yml): test is skipped.
[unskipped][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/skip_tests.yml]: test is skipped because it is not in type filter list: spack
[executors_vars_env_declaration][/home/docs/checkouts/readthedocs.org/user_builds/
↳buildtest/checkouts/v0.11.0/tutorials/script/multiple_executors.yml]: test is skipped
↳because it is not in type filter list: spack
[run_only_macos_distro][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/run_only_distro.yml]: test is skipped because it is not in
↳type filter list: spack
[run_only_linux_distro][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/run_only_distro.yml]: test is skipped because it is not in
↳type filter list: spack
[hello_c][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/compilers/gnu_hello_c.yml]: test is skipped because it is not in type filter
↳list: spack
[string_tag][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/tags_example.yml]: test is skipped because it is not in type filter list:
↳spack
[list_of_strings_tags][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/tags_example.yml]: test is skipped because it is not in
↳type filter list: spack
[exit1_fail][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/pass_returncode.yml]: test is skipped because it is not in type filter
↳list: spack
[exit1_pass][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/pass_returncode.yml]: test is skipped because it is not in type filter
↳list: spack
[returncode_list_mismatch][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/pass_returncode.yml]: test is skipped because it is not in
↳type filter list: spack
[returncode_int_match][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/pass_returncode.yml]: test is skipped because it is not in
↳type filter list: spack
[bash_login_shebang][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/shebang.yml]: test is skipped because it is not in type
↳filter list: spack
```

(continues on next page)

(continued from previous page)

```

[bash_nonlogin_shebang][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/shebang.yml]: test is skipped because it is not in type_
↳filter list: spack
[circle_area][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/python-shell.yml]: test is skipped because it is not in type filter list:_
↳spack
[selinux_disable][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/tutorials/selinux.yml]: test is skipped because it is not in type filter list:_
↳spack
[foo_bar][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/maintainers_example.yml]: test is skipped because it is not in type filter_
↳list: spack
[sleep][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/sleep.yml]: test is skipped because it is not in type filter list: spack
[status_regex_pass][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/tutorials/status_regex.yml]: test is skipped because it is not in type filter_
↳list: spack
[status_regex_fail][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/tutorials/status_regex.yml]: test is skipped because it is not in type filter_
↳list: spack
[run_only_platform_darwin][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/run_only_platform.yml]: test is skipped because it is not_
↳in type filter list: spack
[run_only_platform_linux][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/run_only_platform.yml]: test is skipped because it is not_
↳in type filter list: spack
[add_numbers][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/add_numbers.yml]: test is skipped because it is not in type filter list:_
↳spack
[variables_bash][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/tutorials/vars.yml]: test is skipped because it is not in type filter list: spack
[status_returncode_by_executors][/home/docs/checkouts/readthedocs.org/user_builds/
↳buildtest/checkouts/v0.11.0/tutorials/script/status_by_executors.yml]: test is skipped_
↳because it is not in type filter list: spack
[vecadd_gnu][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↳0/tutorials/compilers/vecadd.yml]: test is skipped because it is not in type filter_
↳list: spack
[executors_sbatach_declaration][/home/docs/checkouts/readthedocs.org/user_builds/
↳buildtest/checkouts/v0.11.0/tutorials/script/executor_scheduler.yml]: test is skipped_
↳because it is not in type filter list: spack
[default_status_regex][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compilers/compiler_status_regex.yml]: test is skipped_
↳because it is not in type filter list: spack
[override_status_regex][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/compilers/compiler_status_regex.yml]: test is skipped_
↳because it is not in type filter list: spack
Valid Buildsspecs: 47
Invalid Buildsspecs: 5
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_regex.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳compilers/envvar_override.yml: VALID

```

(continues on next page)

(continued from previous page)

```

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/compiler_exclude.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_test_specs.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/pre_post_cmds.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ executor_regex_script.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ runtime_status_test.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ metrics_variable.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/pre_post_build_run.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ csh_shell_examples.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ root_user.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/gnu_hello_fortran.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/custom_run.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ hello_world.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ environment.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/metrics_openmp.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ shell_examples.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/openmp_hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ skip_tests.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/multiple_executors.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_distro.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/gnu_hello_c.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ tags_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/remove_environment_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/concretize_m4.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/install_zlib.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_sbatch.yml: VALID

```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_test.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ shebang.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ python-shell.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ selinux.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/spack_multiple_executor_sbatch.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ maintainers_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ sleep.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ status_regex.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ run_only_platform.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ add_numbers.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_create_directory.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/status_by_executors.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/vecadd.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/mirror_example.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_create_manifest.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/executor_scheduler.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ compilers/compiler_status_regex.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ invalid_executor.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ burstbuffer_datawarp_executors.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ invalid_tags.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ invalid_buildspec_section.yml: INVALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ spack/env_install.yml: INVALID
```

Total builder objects created: 14

builders: [spack_test_results_specs_format/62ab8de8, run_pre_post_commands/e870a840, [remove_environment_automatically/e5bfb39a](#), [remove_environment_explicit/a227c364](#), [concretize_m4_in_spack_env/c1ba6948](#), [install_zlib/cb0fb2c0](#), [spack_sbbatch_example/ed2fdade](#), [spack_test/9cf274a3](#), [spack_sbbatch_multi_executors/ec83a273](#), [spack_sbbatch_multi_executors/dd1358a6](#), [spack_env_directory/08b15b87](#), [add_mirror/b568cec6](#), [add_mirror_in_spack_env/1b308bf1](#), [spack_env_create_from_manifest/84cf5193](#)]

[remove_environment_automatically/e5bfb39a](#), [remove_environment_explicit/a227c364](#) (continued on next page)

[concretize_m4_in_spack_env/c1ba6948](#), [install_zlib/cb0fb2c0](#), [spack_sbbatch_example/ed2fdade](#), [spack_test/9cf274a3](#), [spack_sbbatch_multi_executors/ec83a273](#), [spack_sbbatch_multi_executors/dd1358a6](#), [spack_env_directory/08b15b87](#), [add_mirror/b568cec6](#), [add_mirror_in_spack_env/1b308bf1](#), [spack_env_create_from_manifest/84cf5193](#)]

(continued from previous page)

name	id	description	
↳	buildspecs		
-----	-----	-----	
↳			
↳			
spack_test_results_specs_format	62ab8de8	Run spack test results with spec format	↳
↳		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/spack_test_specs.yml			
run_pre_post_commands	e870a840	Install zlib	↳
↳		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/pre_post_cmds.yml			
remove_environment_automatically	e5bfb39a	remove spack environment automatically	↳
↳ before creating a new environment		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/remove_environment_example.yml			
remove_environment_explicit	a227c364	remove spack environment explicitly using	↳
↳ the 'rm' property		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/remove_environment_example.yml			
concretize_m4_in_spack_env	c1ba6948	Concretize m4 in a spack environment named	↳
↳ m4		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/concretize_m4.yml			
install_zlib	cb0fb2c0	Install zlib	↳
↳		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/install_zlib.yml			
spack_sbatch_example	ed2fdade	sbatch directives can be defined in spack	↳
↳ schema		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/spack_sbatch.yml			
spack_test	9cf274a3	Install bzip2 and run spack test and report	↳
↳ results		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/spack_test.yml			
spack_sbatch_multi_executors	ec83a273	sbatch directives can be defined in spack	↳
↳ schema		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/spack_multiple_executor_sbatch.yml			
spack_sbatch_multi_executors	dd1358a6	sbatch directives can be defined in spack	↳
↳ schema		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/spack_multiple_executor_sbatch.yml			
spack_env_directory	08b15b87	Concretize m4 in a spack environment named	↳
↳ m4		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/env_create_directory.yml			
add_mirror	b568cec6	Declare spack mirror	↳
↳		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/mirror_example.yml			
add_mirror_in_spack_env	1b308bf1	Declare spack mirror in spack environment	↳
↳		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/mirror_example.yml			
spack_env_create_from_manifest	84cf5193	Create spack environment from spack.yaml	↳
↳		/home/docs/checkouts/readthedocs.org/user_builds/	
↳ buildtest/checkouts/v0.11.0/tutorials/spack/env_create_manifest.yml			
+-----+			
Stage: Building Test			

(continues on next page)

(continued from previous page)

+-----+					
name	id	type	executor	tags	
↳testpath					↳
-----+-----+-----+-----+-----+					
↳					↳
↳					↳
↳					↳
spack_test_results_specs_format	62ab8de8	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/spack_test_specs/spack_test_results_specs_format/62ab8de8/spack_test_					↳
↳results_specs_format_build.sh					↳
run_pre_post_commands	e870a840	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/pre_post_cmds/run_pre_post_commands/e870a840/run_pre_post_commands_					↳
↳build.sh					↳
remove_environment_automatically	e5bfb39a	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/remove_environment_example/remove_environment_automatically/e5bfb39a/					↳
↳remove_environment_automatically_build.sh					↳
remove_environment_explicit	a227c364	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/remove_environment_example/remove_environment_explicit/a227c364/					↳
↳remove_environment_explicit_build.sh					↳
concretize_m4_in_spack_env	c1ba6948	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/concretize_m4/concretize_m4_in_spack_env/c1ba6948/concretize_m4_in_					↳
↳spack_env_build.sh					↳
install_zlib	cb0fb2c0	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/install_zlib/install_zlib/cb0fb2c0/install_zlib_build.sh					↳
spack_sbatch_example	ed2fdade	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/spack_sbatch/spack_sbatch_example/ed2fdade/spack_sbatch_example_build.					↳
↳sh					↳
spack_test	9cf274a3	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/spack_test/spack_test/9cf274a3/spack_test_build.sh					↳
spack_sbatch_multi_executors	ec83a273	spack	generic.local.bash	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.bash/spack_multiple_executor_sbatch/spack_sbatch_multi_executors/					↳
↳ec83a273/spack_sbatch_multi_executors_build.sh					↳
spack_sbatch_multi_executors	dd1358a6	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/spack_multiple_executor_sbatch/spack_sbatch_multi_executors/dd1358a6/					↳
↳spack_sbatch_multi_executors_build.sh					↳
spack_env_directory	08b15b87	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/env_create_directory/spack_env_directory/08b15b87/spack_env_directory_					↳
↳build.sh					↳
add_mirror	b568cec6	spack	generic.local.sh	['spack']	↳
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/					↳
↳generic.local.sh/mirror_example/add_mirror/b568cec6/add_mirror_build.sh					↳

(continues on next page)

(continued from previous page)

```
add_mirror_in_spack_env      | 1b308bf1 | spack | generic.local.sh | ['spack'] | ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↪ generic.local.sh/mirror_example/add_mirror_in_spack_env/1b308bf1/add_mirror_in_spack_
↪ env_build.sh
spack_env_create_from_manifest | 84cf5193 | spack | generic.local.sh | ['spack'] | ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↪ generic.local.sh/env_create_manifest/spack_env_create_from_manifest/84cf5193/spack_env_
↪ create_from_manifest_build.sh
```

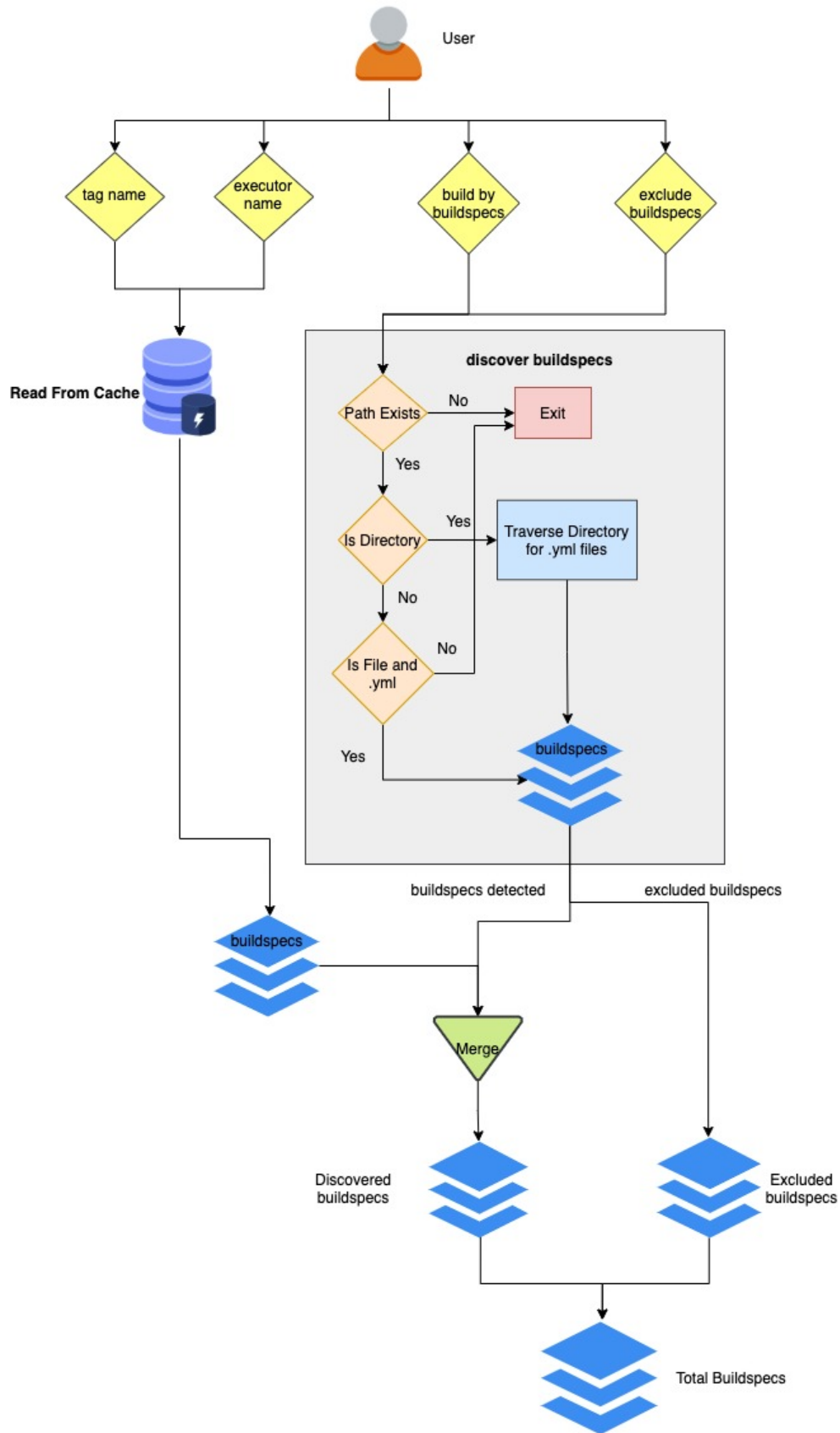
Discover Buildsspecs

Now, let's discuss how buildtest discovers buildsspecs since there are several ways to build buildsspecs.

The buildspec search resolution is described as follows:

- If file or directory specified by `-b` option doesn't exist we exit immediately.
- If buildspec path is a directory, traverse directory recursively to find all `.yaml` extensions
- If buildspec path is a file, check if file extension is not `.yaml`, exit immediately
- If user specifies `--tags` or `--executor` we search in buildspec cache to discover buildsspecs.

Shown below is a diagram on how buildtest discovers buildsspecs. The user can build buildsspecs by `--buildspec`, `-tags`, or `-executor` which will discover the buildsspecs. You can *exclude buildsspecs* using `--exclude` option which is processed after discovering buildsspecs. The excluded buildsspecs are removed from list if found and final list of buildsspecs is processed.



Configure Build Stages

We can control behavior of `buildtest build` command to stop at certain phase using `--stage` option. The `--stage` option accepts `parse` or `build`, which will instruct buildtest to stop at `parse` or `build` phase of the pipeline.

Buildtest will validate all the buildsspecs in the `parse` stage, so you can instruct buildtest to stop at `parse` stage via `--stage=parse`. This can be useful when debugging buildsspecs that are invalid. In this example below, we instruct buildtest to stop after `parse` stage.

```
$ buildtest build -b tutorials/vars.yml --stage=parse
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:21
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/vars.yml --stage=parse

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml: VALID
```

(continues on next page)

(continued from previous page)

```
Total builder objects created: 1
builders: [variables_bash/ac803661]

name          id          description          buildsspecs
-----
variables_bash ac803661 Declare shell variables in bash /home/docs/checkouts/
readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml
```

Likewise, if you want to troubleshoot your test script without running them you can use `--stage=build` which will stop after build phase. This can be used when you are writing builds spec to troubleshoot how test is generated. In this next example, we inform buildtest to stop after build stage.

```
$ buildtest build -b tutorials/vars.yml --stage=build
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:21
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
bin/buildtest build -b tutorials/vars.yml --stage=build

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
vars.yml |
+-----+
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

(continues on next page)

(continued from previous page)

```
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml: VALID
```

```
Total builder objects created: 1
builders: [variables_bash/d2fbee0c]
```

name	id	description	buildspecs
variables_bash	d2fbee0c	Declare shell variables in bash	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml

```

+-----+
| Stage: Building Test |
+-----+

name      | id      | type   | executor      | tags      | testpath
+-----+
variables_bash | d2fbee0c | script | generic.local.bash | ['tutorials'] | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/d2fbee0c/variables_bash_build.sh

```

Invalid Buildsspecs

buildtest will skip any buildsspecs that fail to validate, in that case the test script will not be generated. Here is an example where we have an invalid buildspec.

```
$ buildtest build -b tutorials/invalid_buildspec_section.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:21
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/buildtest build -b tutorials/invalid_buildspec_section.yml
```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪ +-----+
| Discovered Buildsspecs                               ↪
↪
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪invalid_buildspec_section.yml |
+-----+
↪ +-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 0
Invalid Buildsspecs: 1
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪invalid_buildspec_section.yml: INVALID
Unable to create any builder objects

```

buildtest may skip tests from running if builds spec specifies an invalid executor name since buildtest needs to know this in order to delegate test to Executor class responsible for running the test. Here is an example where test failed to run since we provided invalid executor.

```

$ buildtest build -b tutorials/invalid_executor.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:22
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↪python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪bin/buildtest build -b tutorials/invalid_executor.yml

+-----+
| Stage: Discovering Buildsspecs |

```

(continues on next page)

(continued from previous page)

```
+-----+
+-----+
+-----+
| Discovered Buildsspecs |
+-----+
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
+-----+
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 0
Invalid Buildsspecs: 1
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
+-----+
+-----+
Unable to create any builder objects
```

Rebuild Tests

buildtest can rebuild tests using the `--rebuild` option which can be useful if you want to test a particular test multiple times. The rebuild option works across all discovered buildsspecs and create a new test instance (unique id) and test directory path. To demonstrate we will build `tutorials/python-shell.yml` three times using `--rebuild=3`.

```
$ buildtest build -b tutorials/python-shell.yml --rebuild=3
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:22
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
+-----+
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
+-----+
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
+-----+
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
+-----+
+-----+
| Stage: Discovering Buildsspecs |
```

(continues on next page)

(continued from previous page)

```

+-----+
+-----+
+-----+
| Discovered Buildsspecs |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
python-shell.yml |
+-----+
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
python-shell.yml: VALID

Total builder objects created: 3
builders: [circle_area/8e5d3510, circle_area/f5c50ca7, circle_area/629ee0a6]

name          id          description          buildsspecs
-----
circle_area  8e5d3510  Calculate circle of area given a radius  /home/docs/checkouts/
readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-shell.yml
circle_area  f5c50ca7  Calculate circle of area given a radius  /home/docs/checkouts/
readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-shell.yml
circle_area  629ee0a6  Calculate circle of area given a radius  /home/docs/checkouts/
readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-shell.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type  | executor          | tags          |
+-----+
testpath
+-----+
+-----+
circle_area | 8e5d3510 | script | generic.local.python | ['tutorials', 'python'] | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
generic.local.python/python-shell/circle_area/8e5d3510/circle_area_build.sh
circle_area | f5c50ca7 | script | generic.local.python | ['tutorials', 'python'] | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
generic.local.python/python-shell/circle_area/f5c50ca7/circle_area_build.sh

```

(continued from previous page)

```
circle_area | 629ee0a6 | script | generic.local.python | ['tutorials', 'python'] | /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.python/python-shell/circle_area/629ee0a6/circle_area_build.sh

+-----+
| Stage: Running Test |
+-----+

circle_area/8e5d3510: completed with returncode: 0
circle_area/8e5d3510: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/8e5d3510/circle_area.out
circle_area/8e5d3510: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/8e5d3510/circle_area.err
circle_area/f5c50ca7: completed with returncode: 0
circle_area/f5c50ca7: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/f5c50ca7/circle_area.out
circle_area/f5c50ca7: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/f5c50ca7/circle_area.err
circle_area/629ee0a6: completed with returncode: 0
circle_area/629ee0a6: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/629ee0a6/circle_area.out
circle_area/629ee0a6: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/629ee0a6/circle_area.err

-----
Launching test: circle_area
Test ID: 8e5d3510-bbe7-4140-86d7-aa168c6babec
Executor Name: generic.local.python
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510/circle_area_
↳build.sh

-----
Launching test: circle_area
Test ID: f5c50ca7-2ac3-43b0-a59f-1d88eb00f3de
Executor Name: generic.local.python
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.python/python-shell/circle_area/f5c50ca7/circle_area_
↳build.sh

-----
Launching test: circle_area
Test ID: 629ee0a6-94c3-4e62-b05c-964f92c15f0a
Executor Name: generic.local.python
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.python/python-shell/circle_area/629ee0a6/circle_area_
↳build.sh

+-----+
```

(continues on next page)

(continued from previous page)

Stage: Test Summary					
+-----+					
name	id	executor	status	returncode_match	regex_
↪match	runtime_match	returncode	runtime		
+-----+					
↪					
circle_area	f5c50ca7	generic.local.python	PASS	N/A	N/A
↪	N/A	0	0.10842		
circle_area	629ee0a6	generic.local.python	PASS	N/A	N/A
↪	N/A	0	0.040307		
circle_area	8e5d3510	generic.local.python	PASS	N/A	N/A
↪	N/A	0	0.111269		
Passed Tests: 3/3 Percentage: 100.000%					
Failed Tests: 0/3 Percentage: 0.000%					
Writing Logfile to: /tmp/buildtest_z7y7g10g.log					
A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/					
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log					

The rebuild works with all options including: --buildspec, --exclude, --tags and --executor. buildtest will perform rebuild for all discovered tests, for instance in this next example we will discover all tests by tag name **fail** and each test is rebuild twice.

```
$ buildtest build -t fail --rebuild 2
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:22
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↪python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪bin/buildtest build -t fail --rebuild 2

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪-----+
```

(continues on next page)

(continued from previous page)

```
| Discovered Buildsspecs
↪ |
=====
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml |
+-----+
↪ +-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

BREAKDOWN OF BUILDSPECS BY TAGS
-----
Detected Tag Names: ['fail']
+-----+
↪ +-----+
| fail
↪ |
=====
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml |
+-----+
↪ +-----+

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml: VALID

Total builder objects created: 8
builders: [exit1_fail/0de2fede, exit1_pass/bdda64f7, returncode_list_mismatch/3f3f660f,
↪returncode_int_match/202a2a4d, exit1_fail/69a9a391, exit1_pass/6d8369a3, returncode_
↪list_mismatch/866fd8e4, returncode_int_match/e9556b16]

name          id          description
↪buildspecs
-----
↪
↪
exit1_fail    0de2fede  exit 1 by default is FAIL
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml
exit1_pass    bdda64f7  report exit 1 as PASS
↪/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪pass_returncode.yml
```

(continues on next page)

(continued from previous page)

```

returncode_list_mismatch 3f3f660f exit 2 failed since it failed to match returncode 1 ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ pass_returncode.yml
returncode_int_match      202a2a4d exit 128 matches returncode 128 ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ pass_returncode.yml
exit1_fail                69a9a391 exit 1 by default is FAIL ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ pass_returncode.yml
exit1_pass                6d8369a3 report exit 1 as PASS ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ pass_returncode.yml
returncode_list_mismatch 866fd8e4 exit 2 failed since it failed to match returncode 1 ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ pass_returncode.yml
returncode_int_match      e9556b16 exit 128 matches returncode 128 ↵
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ pass_returncode.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags ↵
↪ | testpath
+-----+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+-----+
↪ -----
exit1_fail    | 0de2fede | script | generic.local.sh | ['tutorials', 'fail'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/exit1_fail/0de2fede/exit1_fail_build.sh
exit1_pass    | bdda64f7 | script | generic.local.sh | ['tutorials', 'pass'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/exit1_pass/bdda64f7/exit1_pass_build.sh
returncode_list_mismatch | 3f3f660f | script | generic.local.sh | ['tutorials', 'fail'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/returncode_list_mismatch/3f3f660f/returncode_
↪ list_mismatch_build.sh
returncode_int_match    | 202a2a4d | script | generic.local.sh | ['tutorials', 'pass'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/returncode_int_match/202a2a4d/returncode_int_
↪ match_build.sh
exit1_fail    | 69a9a391 | script | generic.local.sh | ['tutorials', 'fail'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/exit1_fail/69a9a391/exit1_fail_build.sh
exit1_pass    | 6d8369a3 | script | generic.local.sh | ['tutorials', 'pass'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/exit1_pass/6d8369a3/exit1_pass_build.sh
returncode_list_mismatch | 866fd8e4 | script | generic.local.sh | ['tutorials', 'fail'] ↵
↪ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.sh/pass_returncode/returncode_list_mismatch/866fd8e4/returncode_
↪ list_mismatch_build.sh

```

(continues on next page)

(continued from previous page)

```

returncode_int_match      | e9556b16 | script | generic.local.sh | ['tutorials', 'pass']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/returncode_int_match/e9556b16/returncode_int_
↳ match_build.sh

+-----+
| Stage: Running Test |
+-----+

exit1_fail/0de2fede: completed with returncode: 1
exit1_fail/0de2fede: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ fail/0de2fede/exit1_fail.out
exit1_fail/0de2fede: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ fail/0de2fede/exit1_fail.err
returncode_list_mismatch/3f3f660f: completed with returncode: 2
returncode_list_mismatch/3f3f660f: Writing output file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/3f3f660f/returncode_list_mismatch.out
returncode_list_mismatch/3f3f660f: Writing error file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/3f3f660f/returncode_list_mismatch.err
returncode_list_mismatch/3f3f660f: Checking returncode - 2 is matched in list [1, 3]
exit1_fail/69a9a391: completed with returncode: 1
exit1_fail/69a9a391: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ fail/69a9a391/exit1_fail.out
exit1_fail/69a9a391: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ fail/69a9a391/exit1_fail.err
returncode_list_mismatch/866fd8e4: completed with returncode: 2
returncode_list_mismatch/866fd8e4: Writing output file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/866fd8e4/returncode_list_mismatch.out
returncode_list_mismatch/866fd8e4: Writing error file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/866fd8e4/returncode_list_mismatch.err
returncode_list_mismatch/866fd8e4: Checking returncode - 2 is matched in list [1, 3]
exit1_pass/bdda64f7: completed with returncode: 1
exit1_pass/bdda64f7: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/bdda64f7/exit1_pass.out
exit1_pass/bdda64f7: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/bdda64f7/exit1_pass.err
exit1_pass/bdda64f7: Checking returncode - 1 is matched in list [1]
returncode_int_match/202a2a4d: completed with returncode: 128
returncode_int_match/202a2a4d: Writing output file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_int_match/202a2a4d/returncode_int_match.out
returncode_int_match/202a2a4d: Writing error file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_int_match/202a2a4d/returncode_int_match.err

```

(continued from previous page)

```

returncode_int_match/202a2a4d: Checking returncode - 128 is matched in list [128]
exit1_pass/6d8369a3: completed with returncode: 1
exit1_pass/6d8369a3: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/6d8369a3/exit1_pass.out
exit1_pass/6d8369a3: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/6d8369a3/exit1_pass.err
exit1_pass/6d8369a3: Checking returncode - 1 is matched in list [1]
returncode_int_match/e9556b16: completed with returncode: 128
returncode_int_match/e9556b16: Writing output file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_int_match/e9556b16/returncode_int_match.out
returncode_int_match/e9556b16: Writing error file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_int_match/e9556b16/returncode_int_match.err
returncode_int_match/e9556b16: Checking returncode - 128 is matched in list [128]

```

Launching test: exit1_fail

Test ID: 0de2fede-81b4-4f55-83d2-4a85e73f96b6

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/pass_returncode/exit1_fail/0de2fede/exit1_fail_build.sh

Launching test: exit1_pass

Test ID: bdda64f7-c704-4d78-a945-524000e53844

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/bdda64f7/exit1_pass_build.sh

Launching test: returncode_list_mismatch

Test ID: 3f3f660f-d6e1-4341-a240-c7c11234b607

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/3f3f660f/
↳ returncode_list_mismatch_build.sh

Launching test: returncode_int_match

Test ID: 202a2a4d-e874-4b62-a015-0d550ecb8ba4

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/202a2a4d/
↳ returncode_int_match_build.sh

Launching test: exit1_fail

Test ID: 69a9a391-cdce-4269-adb5-b844fb4ad23a

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/pass_returncode/exit1_fail/69a9a391/exit1_fail_build.sh

Launching test: exit1_pass

Test ID: 6d8369a3-badb-47e5-9545-d92bd218c75c

(continues on next page)

(continued from previous page)

Executor Name: generic.local.sh
 Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/6d8369a3/exit1_pass_build.sh

Launching test: returncode_list_mismatch

Test ID: 866fd8e4-48d5-41ca-84ee-e9e2814dda65

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/866fd8e4/returncode_list_mismatch_build.sh

Launching test: returncode_int_match

Test ID: e9556b16-756c-4b87-8b2e-1a45ffaf9b38

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/e9556b16/returncode_int_match_build.sh

+-----+
 | Stage: Test Summary |
 +-----+

name	id	executor	status	returncode_match
regex_match	runtime_match	returncode	runtime	

exit1_fail	0de2fede	generic.local.sh	FAIL	N/A
N/A	N/A	1	0.020173	
exit1_pass	bdda64f7	generic.local.sh	PASS	True
False	False	1	0.019438	
returncode_list_mismatch	3f3f660f	generic.local.sh	FAIL	False
False	False	2	0.019207	
returncode_int_match	202a2a4d	generic.local.sh	PASS	True
False	False	128	0.018725	
exit1_fail	69a9a391	generic.local.sh	FAIL	N/A
N/A	N/A	1	0.017334	
exit1_pass	6d8369a3	generic.local.sh	PASS	True
False	False	1	0.020802	
returncode_list_mismatch	866fd8e4	generic.local.sh	FAIL	False
False	False	2	0.017478	
returncode_int_match	e9556b16	generic.local.sh	PASS	True
False	False	128	0.021757	

Passed Tests: 4/8 Percentage: 50.000%

Failed Tests: 4/8 Percentage: 50.000%

Writing Logfile to: /tmp/buildtest_hldlt9qt.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

The rebuild option expects a range between **1-50**, the `--rebuild=1` is equivalent to running without `--rebuild` option. We set a max limit for rebuild option to avoid system degradation due to high workload.

If you try to exceed this bound you will get an error such as

```
$ buildtest build -b tutorials/pass_returncode.yml --rebuild 51
Traceback (most recent call last):
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
↳ buildtest", line 17, in <module>
    buildtest.main.main()
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/main.py", line 108, in main
    helpfilter=args.helpfilter,
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/cli/build.py", line 525, in __init__
    f"--rebuild {rebuild} exceeds maximum rebuild limit of 50"
buildtest.exceptions.BuildTestError: '--rebuild 51 exceeds maximum rebuild limit of 50'
```

Use Alternate Configuration file

If you want to use an alternate configuration file when building test you can use `buildtest -c <config> build`. buildtest will prefer configuration file on command line over the user configuration (`$HOME/.buildtest/config.yml`). For more details see [Which configuration file does buildtest read?](#).

Keeping Stage Directory

buildtest will create setup the test environment in the *stage* directory where test will be executed. Once test is complete, buildtest will remove the *stage* directory. If you want to preserve the stage directory you can use `buildtest build --keep-stage-dir`, this is only useful if you want to run the test manually

3.3.2 Buildsspecs Interface

Now that we learned how to build tests, in this section we will discuss how one can query a buildspect cache. In buildtest, one can load all buildspects which is equivalent to validating all buildspects with the appropriate schema. Buildtest will ignore all invalid buildspects and store them in a separate file.

Note: `buildtest bc` is an alias for `buildtest buildspect` command.

Finding Buildspects - `buildtest buildspect find`

The `buildtest buildspect find` command is used for finding buildspects from buildspect cache. This command is also used for generating the buildspect cache. Shown below is a list of options for provided for this command.

```
$ buildtest buildspect find --help
usage: buildtest [options] [COMMANDS] buildspect find [-h] [-b] [-e] [--group-by-tags] [--
↳ group-by-executor] [-m] [-mb]
                                                    [-p] [-t] [--filter FILTER] [--
↳ format FORMAT] [--helpfilter]
                                                    [--helpformat] [-n] [--terse] [-r]
↳ [--root ROOT]
```

(continues on next page)

(continued from previous page)

positional arguments:

invalid Show invalid buildsspecs

optional arguments:

-h, --help show this help message and exit
 -r, --rebuild Rebuild buildspec cache and find all buildsspecs again
 --root ROOT Specify root buildsspecs (directory) path to load buildsspecs into.
 ↪ buildspec cache.

filter and format:

filter and format options

--filter FILTER Filter buildspec cache with filter fields in format --filter.
 ↪ key1=val1,key2=val2
 --format FORMAT Format buildspec cache with format fields in format --format.
 ↪ field1,field2,...
 --helpfilter Show Filter fields for --filter option for filtering buildspec.
 ↪ cache output
 --helpformat Show Format fields for --format option for formatting buildspec.
 ↪ cache output

terse:

terse options

-n, --no-header Print output without header in terse output
 --terse Print output in machine readable format

query:

query options to retrieve from buildspec cache

-b, --buildspec Get all buildspec files from cache
 -e, --executors get all unique executors from buildsspecs
 --group-by-tags Group tests by tag name
 --group-by-executor Group tests by executor name
 -m, --maintainers Get all maintainers for all buildsspecs
 -mb, --maintainers-by-buildspecs Show maintainers breakdown by buildsspecs
 -p, --paths print all root buildspec paths
 -t, --tags List all available tags

The buildtest buildspec find command will discover all buildsspecs by recursively searching all .yml extensions. buildtest will validate each buildspec file with the json schema and buildtest will display all valid buildsspecs in the output, all invalid buildsspecs will be stored in a file for post-processing.

```
$ buildtest buildspec find
+-----+-----+-----+-----+
↪ -----+-----+-----+-----+
↪ --+
| name | type | executor | tags |
↪ | description |
↪ |
```

(continues on next page)

```
+=====+=====+=====+=====+
| skip                                | script  | generic.local.bash   | tutorials      |  
↪           | This test is skipped                               |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| unskipped                          | script  | generic.local.bash   | tutorials      |  
↪           | This test is not skipped                           |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| status_regex_pass                  | script  | generic.local.bash   | system         |  
↪           | Pass test based on regular expression               |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| status_regex_fail                  | script  | generic.local.bash   | system         |  
↪           | Pass test based on regular expression               |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| metric_regex_example                | script  | generic.local.sh     | tutorials      |  
↪           | capture result metric from output                   |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| executor_regex_script_schema        | script  | generic.local.(bash|sh) | tutorials      |  
↪           | regular expression test with executor using script schema |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| add_numbers                        | script  | generic.local.bash   | tutorials      |  
↪           | Add X+Y                                             |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| python_hello                       | script  | generic.local.bash   | python         |  
↪           | Hello World python                                 |  
↪ |  
+-----+-----+-----+-----+  
↪---+  
| _bin_sh_shell                      | script  | generic.local.sh     | tutorials      |  
↪           | /bin/sh shell example                             |  
↪ |
```

(continued from previous page)

Test Case	Script	Generic.local.bash	Tutorials
_bin_bash_shell /bin/bash shell example	script	generic.local.bash	tutorials
bash_shell bash shell example	script	generic.local.bash	tutorials
sh_shell sh shell example	script	generic.local.sh	tutorials
shell_options shell options	script	generic.local.sh	tutorials
bash_login_shebang customize shebang line with bash login shell	script	generic.local.bash	tutorials
bash_nonlogin_shebang customize shebang line with default bash (nonlogin) shell	script	generic.local.bash	tutorials
run_only_macos_distro Run test only if distro is darwin.	script	generic.local.bash	mac
run_only_linux_distro Run test only if distro is CentOS.	script	generic.local.bash	mac
metric_variable_assignment capture result metric based on variables and environment variables	script	generic.local.sh	tutorials

(continues on next page)

(continued from previous page)

```

+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| selinux_disable          | script  | generic.local.bash      | tutorials  |
↳   | Check if SELinux is Disabled
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| exit1_fail              | script  | generic.local.sh        | tutorials fail|
↳   | exit 1 by default is FAIL
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| exit1_pass              | script  | generic.local.sh        | tutorials pass|
↳   | report exit 1 as PASS
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| returncode_list_mismatch | script  | generic.local.sh        | tutorials fail|
↳   | exit 2 failed since it failed to match returncode 1
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| returncode_int_match     | script  | generic.local.sh        | tutorials pass|
↳   | exit 128 matches returncode 128
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| sleep                   | script  | generic.local.bash      | tutorials    |
↳   | sleep 2 seconds
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| csh_shell               | script  | generic.local.csh       | tutorials    |
↳   | csh shell example
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| string_tag              | script  | generic.local.bash      | network     |
↳   | tags can be a string
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| list_of_strings_tags     | script  | generic.local.bash      | network ping |
↳   | tags can be a list of strings
↳   |

```

(continues on next page)

(continued from previous page)

```

+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| bash_env_variables          | script  | generic.local.bash      | tutorials  |
↳      | Declare environment variables in default shell (bash)      |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| csh_env_declaration         | script  | generic.local.csh       | tutorials  |
↳      | csh shell example to declare environment variables              |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| tcsh_env_declaration        | script  | generic.local.csh       | tutorials  |
↳      | tcsh shell example to declare environment variables              |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| hello_world                 | script  | generic.local.bash      | tutorials  |
↳      | hello world example                                              |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| foo_bar                     | script  | generic.local.sh        | tutorials  |
↳      | prints variable $FOO                                             |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| variables_bash              | script  | generic.local.bash      | tutorials  |
↳      | Declare shell variables in bash                                  |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| run_only_platform_darwin    | script  | generic.local.python    | tutorials  |
↳      | This test will only run if target platform is Darwin            |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| run_only_platform_linux     | script  | generic.local.python    | tutorials  |
↳      | This test will only run if target platform is Linux              |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| circle_area                 | script  | generic.local.python    | tutorials_ |
↳python      | Calculate circle of area given a radius                          |
↳      |

```

(continues on next page)

(continued from previous page)

```

+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| timelimit_min_max          | script  | generic.local.sh          | tutorials  |
↳      | Run a sleep job for 2 seconds and test pass if its within 1.0-3.0sec |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| timelimit_min              | script  | generic.local.sh          | tutorials  |
↳      | Run a sleep job for 2 seconds and test pass if its exceeds min time of 1.0 |
↳ sec |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| timelimit_max              | script  | generic.local.sh          | tutorials  |
↳      | Run a sleep job for 2 seconds and test pass if it's within max time: 5.0 sec |
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| timelimit_min_fail        | script  | generic.local.sh          | tutorials  |
↳      | This test fails because it runs less than mintime of 10 second
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| timelimit_max_fail        | script  | generic.local.sh          | tutorials  |
↳      | This test fails because it exceeds maxtime of 1.0 second
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| run_only_as_root          | script  | generic.local.bash        | tutorials  |
↳      | This test will only run if current user is root
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| spack_test                 | spack   | generic.local.sh          | spack      |
↳      | Install bzip2 and run spack test and report results
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| spack_env_directory       | spack   | generic.local.sh          | spack      |
↳      | Concretize m4 in a spack environment named m4
↳      |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| spack_sbatch_multi_executors | spack   | generic.local.(sh|bash)   | spack      |
↳      | sbatch directives can be defined in spack schema
↳      |

```

(continues on next page)

(continued from previous page)

+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	run_pre_post_commands	spack	generic.local.sh	spack
↳	Install zlib			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	remove_environment_automatically	spack	generic.local.sh	spack
↳	remove spack environment automatically before creating a new environment			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	remove_environment_explicit	spack	generic.local.sh	spack
↳	remove spack environment explicitly using the 'rm' property			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	spack_test_results_specs_format	spack	generic.local.sh	spack
↳	Run spack test results with spec format			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	concretize_m4_in_spack_env	spack	generic.local.sh	spack
↳	Concretize m4 in a spack environment named m4			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	spack_env_create_from_manifest	spack	generic.local.sh	spack
↳	Create spack environment from spack.yaml			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	spack_sbatch_example	spack	generic.local.sh	spack
↳	sbatch directives can be defined in spack schema			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	add_mirror	spack	generic.local.sh	spack
↳	Declare spack mirror			
↳				
+-----+-----+-----+-----+				
↳	+-----+-----+-----+-----+			
↳	+-----+-----+-----+-----+			
↳	add_mirror_in_spack_env	spack	generic.local.sh	spack
↳	Declare spack mirror in spack environment			
↳				

(continues on next page)

(continues on next page)

(continued from previous page)

```

+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| install_zlib                | spack    | generic.local.sh    | spack    |
↳   | Install zlib
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| executors_sbatch_declaration | script   | generic.local.(bash|sh) | tutorials |
↳   | Declaring env and vars by executors section
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| status_returncode_by_executors | script   | generic.local.(bash|sh) | tutorials |
↳   | define status and metrics per executor type.
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| executors_vars_env_declaration | script   | generic.local.(bash|sh) | tutorials |
↳   | Declaring env and vars by executors section
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| openmp_hello_c_example       | compiler | generic.local.bash    | tutorials |
↳ compile   | OpenMP Hello World C example
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| default_status_regex         | compiler | generic.local.bash    | tutorials |
↳ compile   | Regular expression check in stdout for gcc group
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| override_status_regex        | compiler | generic.local.bash    | tutorials |
↳ compile   | Override regular expression for compiler gcc/10.2.0-37fmsw7
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| override_environmentvars      | compiler | generic.local.bash    | tutorials |
↳ compile   | override default environment variables
↳   |
+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
↳ --+
| custom_run_by_compilers       | compiler | generic.local.bash    | tutorials |
↳ compile   | Customize binary launch based on compiler
↳   |

```

(continues on next page)

(continued from previous page)

+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	vecadd_gnu_exclude	compiler generic.local.bash	tutorials_
→	compile	Vector Addition example with GNU compilers but exclude gcc@10.2.0	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	hello_c	compiler generic.local.bash	tutorials_
→	compile	Hello World C Compilation	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	vecadd_gnu	compiler generic.local.bash	tutorials_
→	compile	Vector Addition example with GNU compiler	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	hello_f	compiler generic.local.bash	tutorials_
→	compile	Hello World Fortran Compilation	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	pre_post_build_run	compiler generic.local.bash	tutorials_
→	compile	example using pre_build, post_build, pre_run, post_run example	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	metrics_variable_compiler	compiler generic.local.bash	tutorials_
→	compile	define metrics with compiler schema	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	show_lsf_user_groups	script generic.local.bash	lsf
→		Show information about all LSF user groups	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	show_host_groups	script generic.local.bash	lsf
→		Show information about host groups using bmgrou	→
→			
+-----+-----+-----+-----+			
→	+-----+-----+-----+-----+		
→	+-----+-----+-----+-----+		
	show_lsf_queues	script generic.local.bash	lsf
→		Show LSF queues	→
→			

(continues on next page)

(continued from previous page)

+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	show_lsf_queues_formatted	script	generic.local.bash	lsf
↪	Show LSF queues with formatted columns			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	show_lsf_queues_current_user	script	generic.local.bash	lsf
↪	Show LSF queues available for current user			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	show_lsf_configuration	script	generic.local.bash	lsf
↪	Show LSF configuration using lsinfo			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	show_lsf_models	script	generic.local.bash	lsf
↪	Show information about host models in LSF cluster			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	show_lsf_resources	script	generic.local.bash	lsf
↪	Show information about LSF resources			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	lsf_version	script	generic.local.bash	lsf
↪	Display lsf version using lsinfo			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	display_lsf_hosts	script	generic.local.bash	lsf
↪	Show all hosts in LSF cluster			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	display_hosts_format	script	generic.local.bash	lsf
↪	Show all hosts with column hostname and status			↪
↪				
+	-----+	-----+	-----+	-----+
↪	-----+	-----+	-----+	-----+
↪	--+			
	bhosts_version	script	generic.local.bash	lsf
↪	display version from bhosts command			↪
↪				

(continues on next page)

(continued from previous page)

+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
nodes_state_reboot	script	generic.local.bash	slurm
↳ Show nodes in REBOOT state			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
nodes_state_allocated	script	generic.local.bash	slurm
↳ Show nodes in ALLOCATED state			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
nodes_state_completing	script	generic.local.bash	slurm
↳ Show nodes in COMPLETING state			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
nodes_state_idle	script	generic.local.bash	slurm
↳ Show nodes in IDLE state			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
node_down_fail_list_reason	script	generic.local.bash	slurm
↳ Show nodes DOWN, DRAINED, FAIL or FAILING and list reason			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
dead_nodes	script	generic.local.bash	slurm
↳ Show non-responding nodes			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
get_partitions	script	generic.local.bash	slurm
↳ Get all slurm partitions			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
sinfo_version	script	generic.local.bash	slurm
↳ get slurm version			↳
↳			
+-----+-----+-----+-----+			
+-----+-----+-----+-----+			
↳ --+			
qsub_version	script	generic.local.sh	cobalt
↳ print version for qsub command			↳
↳			

(continues on next page)

(continued from previous page)

+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	kernel_swapusage	script generic.local.bash	configuration ↪
↪	Retrieve Kernel Swap Usage		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	ulimit_filelock_unlimited	script generic.local.bash	system ↪
↪	Check if file lock is set to unlimited in ulimits		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	ulimit_cputime_unlimited	script generic.local.bash	system ↪
↪	Check if cputime is set to unlimited in ulimits		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	ulimit_stacksize_unlimited	script generic.local.bash	system ↪
↪	Check if stack size is set to unlimited in ulimits		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	ulimit_vmsize_unlimited	script generic.local.bash	system ↪
↪	Check virtual memory size and check if its set to unlimited		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	ulimit_filesdescriptor_4096	script generic.local.bash	system ↪
↪	Check if open file descriptors limit is set to 4096		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	ulimit_max_user_process_2048	script generic.local.bash	system ↪
↪	Check max number of user process limit is set to 2048		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	runImage	script generic.local.bash	containers ↪
↪singularity	run container docker://godlovedc/lolcow		
↪			
+-----+-----+-----+-----+			
↪	+-----+-----+-----+-----+		
↪	+-----+-----+-----+-----+		
↪	build_sif_from_dockerimage	script generic.local.bash	containers ↪
↪singularity	build sif image from docker image docker://godlovedc/lolcow		
↪			

↪(continues on next page)

(continues on next page)

(continued from previous page)

```
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
| build_sandbox_image          | script  | generic.local.bash          | containers_
↳singularity | build sandbox image from docker image docker://godlovedc/lolcow
↳
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
| build_remoteimages           | script  | generic.local.bash          | containers_
↳singularity | build remote hosted image from AWS
↳
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
| pullImage_dockerhub          | script  | generic.local.bash          | containers_
↳singularity | Pull image docker://godlovedc/lolcow from DockerHub
↳
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
| pullImage_sylabscloud        | script  | generic.local.bash          | containers_
↳singularity | Pull image library://alpine:latest from Sylabs Cloud
↳
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
| pullImage_shub               | script  | generic.local.bash          | containers_
↳singularity | Pull image shub://vsoch/singularity-images from SingularityHub
↳
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
| inspect_image                | script  | generic.local.bash          | containers_
↳singularity | Inspect image via 'singularity inspect'
↳
+-----+-----+-----+-----+
↳-----+-----+-----+-----+
↳--+
```

buildtest will load all discovered buildsspecs in a cache file (JSON) which is created upon running `buildtest buildspec find`. Any subsequent runs will read from cache and update if any new buildsspecs are added. If you make changes to buildspec you should rebuild the buildspec cache by running:

```
$ buildtest buildspec find --rebuild
```

If you want to find all buildspec files in cache you can run `buildtest buildspec find --buildspec`. Shown below is an example output.

```
$ buildtest buildspec find --buildspec
```

```
+-----+
↳-----+
```

(continues on next page)

(continued from previous page)

```

| buildspecs
|
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| ↪ skip_tests.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| ↪ status_regex.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| ↪ metrics_regex.yml |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| ↪ executor_regex_script.yml |
+-----+
|
+-----+
...

```

The buildtest `buildspec find --paths` will display a list of root directories buildtest will search for buildspecs when running `buildtest buildspec find`. One can define these directories in the configuration file or pass them via command line.

```

$ buildtest buildspec find --paths
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
| ↪ tests

```

buildtest will search buildspecs in *buildspecs root* defined in your configuration, which is a list of directory paths to search for buildspecs. If you want to load buildspecs from a directory path, you can specify a directory via `--root` option in the format: `buildtest buildspec find --root <path> --rebuild`. buildtest will load all valid buildspecs into cache and ignore the rest. It's important to add `--rebuild` if you want to regenerate buildspec cache.

Filtering buildspec

Once you have a buildspec cache, we can query the buildspec cache for certain attributes. When you run **buildtest buildspec find** it will report all buildspecs from cache which can be difficult to process. Therefore, we have a filter option (`--filter`) to restrict our search. Let's take a look at the available filter fields that are acceptable with filter option.

```

$ buildtest buildspec find --helpfilter
Field      Description      Type
-----
buildspec  Filter tests by buildspec  FILE
executor   Filter by executor name    STRING
tags       Filter by tag name         STRING
type       Filter by schema type      STRING

```

The `--filter` option expects an arguments in **key=value** format as follows:

```
buildtest buildspect find --filter key1=value1,key2=value2,key3=value3
```

We can filter buildspect cache by tags=fail which will query all tests with associated tag field in test.

```
$ buildtest buildspect find --filter tags=fail
+-----+-----+-----+-----+-----+
| name                | type  | executor      | tags          | description      |
+-----+-----+-----+-----+-----+
| exit1_fail          | script | generic.local.sh | tutorials fail | exit 1 by default is FAIL |
+-----+-----+-----+-----+-----+
| returncode_list_mismatch | script | generic.local.sh | tutorials fail | exit 2 failed since it failed to match returncode 1 |
+-----+-----+-----+-----+-----+
```

In addition, we can query buildspects by schema type using type property. In this example we query all tests by type property

```
$ buildtest buildspect find --filter type=script
+-----+-----+-----+-----+-----+
| name                | type  | executor      | tags          | description      |
+-----+-----+-----+-----+-----+
| skip                | script | generic.local.bash | tutorials     | This test is skipped |
+-----+-----+-----+-----+-----+
| unskipped           | script | generic.local.bash | tutorials     | This test is not skipped |
+-----+-----+-----+-----+-----+
| status_regex_pass   | script | generic.local.bash | system        | Pass test based on regular expression |
+-----+-----+-----+-----+-----+
| status_regex_fail   | script | generic.local.bash | system        | Pass test based on regular expression |
+-----+-----+-----+-----+-----+
| metric_regex_example | script | generic.local.sh   | tutorials     | capture result metric from output |
+-----+-----+-----+-----+-----+
| executor_regex_script_schema | script | generic.local.(bash|sh) | tutorials     | regular expression test with executor using script schema |
+-----+-----+-----+-----+-----+
| add_numbers         | script | generic.local.bash | tutorials     | Add X+Y |
+-----+-----+-----+-----+-----+
```

(continues on next page)

```

+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+
| python_hello          | script | generic.local.bash    | python          | ↪
↪ | Hello World python  |        |                        |                 |
+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+
| _bin_sh_shell         | script | generic.local.sh      | tutorials       | ↪
↪ | /bin/sh shell example |        |                        |                 |
+-----+-----+-----+-----+
↪ +-----+-----+-----+-----+
...

```

```
$ buildtest buildspect find --filter tags=tutorials,executor=generic.local.sh,type=script
+-----+-----+-----+-----+-----+
| name                                | type  | executor      | tags      | description |
+-----+-----+-----+-----+-----+
| metric_regex_example                | script | generic.local.sh | tutorials | capture    |
| result metric from output          |        |                  |          |            |
+-----+-----+-----+-----+-----+
| _bin_sh_shell                       | script | generic.local.sh | tutorials | /bin/sh    |
| shell example                      |        |                  |          |            |
+-----+-----+-----+-----+-----+
| sh_shell                           | script | generic.local.sh | tutorials | sh shell   |
| example                            |        |                  |          |            |
+-----+-----+-----+-----+-----+
| shell_options                       | script | generic.local.sh | tutorials | shell      |
| options                            |        |                  |          |            |
+-----+-----+-----+-----+-----+
| metric_variable_assignment           | script | generic.local.sh | tutorials | capture    |
| result metric based on variables and environment variable |        |                  |          |            |
+-----+-----+-----+-----+-----+
| exit1_fail                          | script | generic.local.sh | tutorials fail | exit 1 by  |
| default is FAIL                    |        |                  |          |            |
+-----+-----+-----+-----+-----+
| exit1_pass                          | script | generic.local.sh | tutorials pass | report exit |
| 1 as PASS                          |        |                  |          |            |
+-----+-----+-----+-----+-----+
| returncode_list_mismatch             | script | generic.local.sh | tutorials fail | exit 2     |
| failed since it failed to match returncode 1 |        |                  |          |            |
+-----+-----+-----+-----+-----+
```

(continued from previous page)

returncode_int_match	script generic.local.sh tutorials pass exit 128	
↳ matches returncode 128		
+-----+-----+-----+-----+-----+-----+		
↳		
foo_bar	script generic.local.sh tutorials	prints
↳ variable \$FOO		
+-----+-----+-----+-----+-----+-----+		
↳		
timelimit_min_max	script generic.local.sh tutorials	Run a sleep
↳ job for 2 seconds and test pass if its within 1.0-3.0sec		
+-----+-----+-----+-----+-----+-----+		
↳		
timelimit_min	script generic.local.sh tutorials	Run a sleep
↳ job for 2 seconds and test pass if its exceeds min time of 1.0 sec		
+-----+-----+-----+-----+-----+-----+		
↳		
timelimit_max	script generic.local.sh tutorials	Run a sleep
↳ job for 2 seconds and test pass if it's within max time: 5.0 sec		
+-----+-----+-----+-----+-----+-----+		
↳		
timelimit_min_fail	script generic.local.sh tutorials	This test
↳ fails because it runs less than mintime of 10 second		
+-----+-----+-----+-----+-----+-----+		
↳		
timelimit_max_fail	script generic.local.sh tutorials	This test
↳ fails because it exceeds maxtime of 1.0 second		
+-----+-----+-----+-----+-----+-----+		
↳		

We can filter output of buildspec cache by buildspec using `--filter buildspec=<path>` which expects a path to buildspec file. The buildspec must be in the cache and file path must exist in order to fetch the result. The path can be absolute or relative path.

In this next example, we will filter cache by file `tutorials/pass_returncode.yml` and use `--format name,buildspec` to format columns. The `--format buildspec` will show full path to buildspec and `name` refers to name of test. For more details on `-format` see [Format buildspec cache](#).

```
$ buildtest buildspec find --filter buildspec=tutorials/pass_returncode.yml --format
↳ name,buildspec
+-----+-----+-----+-----+-----+-----+
| name                | buildspec                                     |
|-----|-----|
| exit1_fail          | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/pass_returncode.yml |
+-----+-----+-----+-----+-----+-----+
| exit1_pass          | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/pass_returncode.yml |
+-----+-----+-----+-----+-----+-----+
| returncode_list_mismatch | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/pass_returncode.yml |
```

(continues on next page)

(continued from previous page)

```

+-----+-----+
| returncode_int_match      | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
| checkouts/v0.11.0/tutorials/pass_returncode.yml |
+-----+-----+

```

Format buildspec cache

We have seen how one can filter buildspecs, but we can also configure which columns to display in the output of **buildtest buildspec find**. By default, we show a pre-selected format fields in the output, however there are more format fields available that can be configured at the command line.

The format fields are specified in comma separated format such as `buildtest buildspec find --format <field1>,<field2>,...`. You can see a list of all format fields by `--helpformat` option as shown below

```

$ buildtest buildspec find --helpformat
Field      Description
-----
buildspec  Display name of buildspec file
description Show description of test
executor   Display 'executor' property in test
name       Display name of test
tags       Display 'tag' property in test
type       Display 'type' property in test

```

In the next example, we utilize `--format` field with `--filter` option to show how format fields affect table columns. `buildtest` will display the table in order of format fields specified in command line.

```

$ buildtest buildspec find --format name,description,buildspec --filter tags=tutorials,
| executor=generic.local.sh
+-----+-----+
| name          | description                                     |
|               | buildspec                                       |
+-----+-----+
| metric_regex_example | capture result metric from output
|               | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
| checkouts/v0.11.0/tutorials/metrics_regex.yml |
+-----+-----+
| _bin_sh_shell | /bin/sh shell example
|               | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
| checkouts/v0.11.0/tutorials/shell_examples.yml |
+-----+-----+
| sh_shell      | sh shell example
|               | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
| checkouts/v0.11.0/tutorials/shell_examples.yml |

```

(continues on next page)

(continued from previous page)

```

+-----+
+-----+
+-----+
| shell_options          | shell options                                     |
+-----+
|                         | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/shell_examples.yml      |
+-----+
+-----+
+-----+
| metric_variable_assignment | capture result metric based on variables and environment.
+-----+
| variable                | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/metrics_variable.yml    |
+-----+
+-----+
+-----+
| exit1_fail              | exit 1 by default is FAIL                         |
+-----+
|                         | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/pass_returncode.yml      |
+-----+
+-----+
+-----+
| exit1_pass              | report exit 1 as PASS                             |
+-----+
|                         | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/pass_returncode.yml      |
+-----+
+-----+
+-----+
| returncode_list_mismatch | exit 2 failed since it failed to match returncode 1
+-----+
|                         | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/pass_returncode.yml      |
+-----+
+-----+
+-----+
| returncode_int_match     | exit 128 matches returncode 128                   |
+-----+
|                         | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/pass_returncode.yml      |
+-----+
+-----+
+-----+
| foo_bar                  | prints variable $FOO                               |
+-----+
|                         | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/maintainers_example.yml  |
+-----+
+-----+
+-----+
| timelimit_min_max        | Run a sleep job for 2 seconds and test pass if its
+-----+
| 1.0-3.0sec              | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
+-----+
|                         | checkouts/v0.11.0/tutorials/runtime_status_test.yml  |
+-----+
+-----+
+-----+
| timelimit_min            | Run a sleep job for 2 seconds and test pass if its
+-----+
| exceeds min time of 1.0 sec | /home/docs/checkouts/readthedocs.org/user_builds/
+-----+
|                         | buildtest/checkouts/v0.11.0/tutorials/runtime_status_test.yml |
+-----+

```

(continued from previous page)

```

+-----+
↪-----+
↪-----+
| timelimit_max          | Run a sleep job for 2 seconds and test pass if it's_
↪within max time: 5.0 sec | /home/docs/checkouts/readthedocs.org/user_builds/
↪buildtest/checkouts/v0.11.0/tutorials/runtime_status_test.yml |
+-----+
↪-----+
↪-----+
| timelimit_min_fail     | This test fails because it runs less than mintime of 10_
↪second                  | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/tutorials/runtime_status_test.yml |
+-----+
↪-----+
↪-----+
| timelimit_max_fail     | This test fails because it exceeds maxtime of 1.0 second _
↪                        | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/tutorials/runtime_status_test.yml |
+-----+
↪-----+
↪-----+

```

buildtest makes use of python library named [tabulate](#) to generate these tables which are found in commands line like `buildtest buildspect find` and `buildtest report`.

Querying buildspect tags

If you want to retrieve all unique tags from all buildspects you can run `buildtest buildspect find --tags`. This can be useful if you want to know available tags in your buildspect cache.

```

$ buildtest buildspect find --tags
+-----+
| Tags    |
+=====+
| network |
+-----+
| filesystem |
+-----+
| cobalt   |
+-----+
| containers |
+-----+
| spack    |
+-----+
| lsf      |
+-----+
| compile  |
+-----+
| fail     |
+-----+
| system   |

```

(continues on next page)

(continued from previous page)

```

+-----+
| ping      |
+-----+
| python    |
+-----+
| singularity |
+-----+
| pass      |
+-----+
| mac       |
+-----+
| tutorials  |
+-----+
| ssh       |
+-----+
| storage    |
+-----+
| slurm      |
+-----+
| configuration |
+-----+

```

In addition, buildtest can group tests by tags via `buildtest buildspect find --group-by-tags` which can be useful if you want to know which tests get executed when running `buildtest build --tags`. The output is grouped by tag names, followed by name of test and description.

```

$ buildtest buildspect find --group-by-tags
+-----+-----+
| tags      | name          |
+=====+=====+
| tutorials | skip          |
+-----+-----+
| tutorials | unskipped     |
+-----+-----+
| tutorials | metric_regex_example |
+-----+-----+
| tutorials | executor_regex_script_schema |
+-----+-----+
| tutorials | add_numbers   |
+-----+-----+
| tutorials | _bin_sh_shell |
+-----+-----+
| tutorials | _bin_bash_shell |
+-----+-----+
| tutorials | bash_shell    |
+-----+-----+
| tutorials | sh_shell      |
+-----+-----+
| tutorials | shell_options |
+-----+-----+
| tutorials | bash_login_shebang |
+-----+-----+

```

(continues on next page)

(continued from previous page)

tutorials	bash_nonlogin_shebang	
+-----+	+-----+	+-----+
tutorials	metric_variable_assignment	
+-----+	+-----+	+-----+
tutorials	selinux_disable	
+-----+	+-----+	+-----+
tutorials	exit1_fail	
+-----+	+-----+	+-----+
tutorials	exit1_pass	
+-----+	+-----+	+-----+
tutorials	returncode_list_mismatch	
+-----+	+-----+	+-----+
tutorials	returncode_int_match	
+-----+	+-----+	+-----+
tutorials	sleep	
+-----+	+-----+	+-----+
...		

Querying buildspec executor

If you want to know all executors in your buildspec cache use the `buildtest buildspec find --executors` command. This can be useful when you want to build by executors (`buildtest build --executor`).

```
$ buildtest buildspec find --executors
```

```
+-----+
| executors |
+=====+
| generic.local.bash |
+-----+
| generic.local.(bash|sh) |
+-----+
| generic.local.csh |
+-----+
| generic.local.(sh|bash) |
+-----+
| generic.local.sh |
+-----+
| generic.local.python |
+-----+
```

Similar to `--group-by-tags`, buildtest has an option to group tests by executor using `--group-by-executor` option. This will show tests grouped by executor, name of test and test description. Shown below is an example output.

```
$ buildtest buildspec find --group-by-executor
```

```
+-----+-----+
| executor | name |
+=====+=====+
| generic.local.bash | skip |
+-----+-----+
| generic.local.bash | unskipped |
+-----+-----+
```

(continues on next page)

(continued from previous page)

generic.local.bash	status_regex_pass	
+-----+	+-----+	+-----+
generic.local.bash	status_regex_fail	
+-----+	+-----+	+-----+
generic.local.bash	add_numbers	
+-----+	+-----+	+-----+
generic.local.bash	python_hello	
+-----+	+-----+	+-----+
generic.local.bash	_bin_bash_shell	
+-----+	+-----+	+-----+
generic.local.bash	bash_shell	
+-----+	+-----+	+-----+
generic.local.bash	bash_login_shebang	
+-----+	+-----+	+-----+
generic.local.bash	bash_nonlogin_shebang	
+-----+	+-----+	+-----+
generic.local.bash	run_only_macos_distro	
+-----+	+-----+	+-----+
generic.local.bash	run_only_linux_distro	
+-----+	+-----+	+-----+
generic.local.bash	selinux_disable	
+-----+	+-----+	+-----+
generic.local.bash	sleep	
+-----+	+-----+	+-----+
...		

Query Maintainers

When you are writing your buildspecs, you can specify the `maintainers` field to assign authors to buildspecs. `buildtest` can query the maintainers from the cache once buildspecs are loaded. You can retrieve all maintainers using `--maintainers` option or `-m` short option. In this example, we show all maintainers for buildspecs in buildspec cache

```
$ buildtest buildspect find --maintainers
+-----+
| maintainers |
+=====+
| @shahzebsiddiqui |
+-----+
| @johndoe |
+-----+
| @bobsmith |
+-----+
```

If you want to see a breakdown of maintainers by buildspec file you can use `--maintainers-by-buildspecs` or `-mb` short option. This can be useful to get correlation between maintainers and the buildspec file.

```
$ buildtest buildspect find -mb
+-----+-----+
| maintainers | buildspect |
+-----+-----+
(continues on next page)
```

(continued from previous page)

```

=====+=====
| @shahzebsiddiqui | ['/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/hello_world.yml', '/home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/general_tests/configuration/ulimits.yml'] |
+-----+-----
↳-----+
↳-----+
| @johndoe          | ['/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/maintainers_example.yml'] |
↳-----+
+-----+-----
↳-----+
↳-----+
| @bobsmith         | ['/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/maintainers_example.yml'] |
↳-----+
+-----+-----
↳-----+
↳-----+

```

Terse Output

You can use the `--terse` option to print output of `buildtest buildspec find` in terse format that can be useful if you want to parse content of file. In example below, we will print output of tags in terse format, the first entry `tags` is the header followed by list of unique tags. The `--no-header` option can be used to disable printing of header title.

```

$ buildtest buildspec find -t --terse
tag
network
filesystem
cobalt
containers
spack
lsf
compile
fail
system
ping
python
singularity
pass
mac
tutorials
ssh
storage
slurm
configuration

```

Invalid Buildsspecs - buildtest buildspec find invalid

buildtest will store invalid buildsspecs in the cache file which can be retrieved using `buildtest buildspec find invalid`. buildtest will attempt to parse each buildspec and store error message for every buildspec. If you run without any options it will report a list of invalid buildsspecs as shown below

```
$ buildtest buildspec find invalid
+-----+
| buildspecs                                     |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| invalid_executor.yml                          |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| invalid_tags.yml                             |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| burstbuffer_datawarp_executors.yml          |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| invalid_buildspec_section.yml               |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
| spack/env_install.yml                      |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
| tests/sched/pbs/hostname.yml               |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
| tests/sched/pbs/batch.yml                  |
+-----+
```

If you want to see error messages for each buildspec you can pass the `-e` or `--error` option which will display output of each buildspec followed by error message.

```
$ buildtest buildspec find invalid -e
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
invalid_executor.yml

"/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
invalid_executor.yml]: Unable to find executor: badexecutor in ['generic.local.bash',
'generic.local.sh', 'generic.local.csh', 'generic.local.python']"
```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_tags.yml
```

['network', 'network'] is not valid under any of the given schemas

Failed validating 'oneOf' in schema['properties']['tags']:

```
{'oneOf': [{'type': 'string'},
            {'$ref': '#/definitions/list_of_strings'}]}
```

On instance['tags']:

```
['network', 'network']
```

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳burstbuffer_datawarp_executors.yml
```

'create_burst_buffer_multiple_executors' is too long

Failed validating 'maxLength' in schema['properties']['buildspecs']['propertyNames']:

```
{'maxLength': 32, 'pattern': '^[A-Za-z_.][A-Za-z0-9_.]*$'}
```

On instance['buildspecs']:

```
'create_burst_buffer_multiple_executors'
```

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_buildspec_section.yml
```

'[/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳invalid_buildspec_section.yml]: type badscript is not known to buildtest.'

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳spack/env_install.yml
```

Additional properties are not allowed ('option' was unexpected)

Failed validating 'additionalProperties' in schema['properties']['spack']['properties']['
↳install']:

```
{'additionalProperties': False,
 'description': 'Install spack packages using ``spack install`` ',
 'command',
 'properties': {'options': {'description': 'Pass options to ``spack '
                                'install`` command',
```

(continues on next page)

(continued from previous page)

```

        'type': 'string'},
        'specs': {'$ref': 'definitions.schema.json#/definitions/list_of_
↳ strings',
                    'description': 'List of specs to install '
                                   'using ``spack install`` '
                                   'command'}},
        'type': 'object'}

On instance['spack']['install']:
    {'option': '--keep-prefix'}

-----

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/sched/pbs/hostname.yml

-----

"/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/sched/pbs/hostname.yml]: Unable to find executor: generic.pbs.workq in ['generic.
↳ local.bash', 'generic.local.sh', 'generic.local.csh', 'generic.local.python']"

-----

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/sched/pbs/batch.yml

-----

"/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/sched/pbs/batch.yml]: Unable to find executor: generic.pbs.workq in ['generic.
↳ local.bash', 'generic.local.sh', 'generic.local.csh', 'generic.local.python']"

-----

```

Cache Summary - buildtest buildspec summary

The `buildtest buildspec summary` command can be used to provide a summary of the buildspec cache. This command can be used assuming your cache is built via `buildtest buildspec find`. Shown below is a summary of the cache file.

```

$ buildtest buildspec summary
Reading Buildspeg Cache File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/buildspecs/cache.json

Search Paths: ['/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/tutorials', '/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests']
Total Valid Buildspegcs: 66
Total Invalid Buildspegcs: 7
Total Unique Tags: 19
Total Unique Executors: 6
Total Maintainers: 3

```

(continues on next page)

(continued from previous page)

```

Unique Tags: ['network', 'filesystem', 'cobalt', 'containers', 'spack', 'lsf', 'compile
↳ ', 'fail', 'system', 'ping', 'python', 'singularity', 'pass', 'mac', 'tutorials', 'ssh
↳ ', 'storage', 'slurm', 'configuration']
Unique Executors: ['generic.local.bash', 'generic.local.(bash|sh)', 'generic.local.csh',
↳ 'generic.local.(sh|bash)', 'generic.local.sh', 'generic.local.python']
Unique Maintainers: ['@shahzebsiddiqui', '@johndoe', '@bobsmith']

```

Tag Breakdowns

name	total
tutorials	49
system	9
python	2
mac	2
fail	2
pass	2
network	2
ping	1
spack	13
compile	11
lsf	12
slurm	17
cobalt	7
filesystem	1
storage	1
ssh	1
configuration	1
containers	8
singularity	8

(continues on next page)

(continued from previous page)

Executor Breakdowns

name	total
generic.local.bash	78
generic.local.sh	34
generic.local.(bash sh)	4
generic.local.csh	3
generic.local.python	3
generic.local.(sh bash)	1

Test Breakdown by buildsspecs

buildspec	total
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/skip_tests.yml	2
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/status_regex.yml	2
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/metrics_regex.yml	1
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/executor_regex_script.yml	1
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/add_numbers.yml	1

(continues on next page)

(continued from previous page)

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪python-hello.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪shell_examples.yml	5
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪shebang.yml	2
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪run_only_distro.yml	2
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪metrics_variable.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪selinux.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪pass_returncode.yml	4
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪sleep.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪csh_shell_examples.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪tags_example.yml	2
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪environment.yml	3
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪hello_world.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪maintainers_example.yml	1
+-----+-----+	

(continues on next page)

(continued from previous page)

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪vars.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪run_only_platform.yml	2
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪python-shell.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪runtime_status_test.yml	5
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪root_user.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/spack_test.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/env_create_directory.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/spack_multiple_executor_sbatch.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/pre_post_cmds.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/remove_environment_example.yml	2
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/spack_test_specs.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/concretize_m4.yml	1
+-----+-----+	
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪spack/env_create_manifest.yml	1
+-----+-----+	

(continues on next page)

(continued from previous page)

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳spack/spack_sbatch.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳spack/mirror_example.yml	2
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳spack/install_zlib.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳script/executor_scheduler.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳script/status_by_executors.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳script/multiple_executors.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/openmp_hello.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/compiler_status_regex.yml	2
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/envvar_override.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/custom_run.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/compiler_exclude.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/gnu_hello_c.yml	1
-----+-----	
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↳compilers/vecadd.yml	1
-----+-----	

(continues on next page)

(continued from previous page)

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪compilers/gnu_hello_fortran.yml	1
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪compilers/pre_post_build_run.yml	1
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/ ↪compilers/metrics_openmp.yml	1
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/lsf/bugroup.yml	1
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/lsf/bmggroups.yml	1
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/lsf/bqueues.yml	3
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/lsf/lsinfo.yml	4
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/lsf/bhosts.yml	3
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/slurm/squeue.yml	2
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/slurm/sacctmgr.yml	4
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/slurm/scontrol.yml	2
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/slurm/sinfo.yml	9
-----+-----	
↪-----+-----	
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_ ↪tests/sched/cobalt/commands.yml	7
-----+-----	
↪-----+-----	

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/disk_usage.yml | 1 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/systemd-default-target.yml | 1 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ssh_localhost.yml | 1 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/kernel_state.yml | 1 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/configuration/ulimits.yml | 6 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/containers/singularity/run.yml | 1 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/containers/singularity/build.yml | 3 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/containers/singularity/pull.yml | 3 |
+-----+
↳ -----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/containers/singularity/inspect.yml | 1 |
+-----+
↳ -----+
```

Validate Buildsspecs - buildtest buildspec validate

buildtest can validate buildsspecs through the `buildtest buildspec validate` command which provides analogous options for `buildtest build` for selecting buildsspecs such as `-b`, `-e`, `-t` and `-e`. This command can be used to validate buildsspecs with the JSON Schema which can be useful if you are writing a buildspec and want to validate the buildspec without running the test.

Shown below are the available command options.

```
$ buildtest buildspec validate --help
usage: buildtest [options] [COMMANDS] buildspec validate [-h] [-b BUILDSPEC] [-x_
↳ EXCLUDE] [-e EXECUTOR] [-t TAG]

optional arguments:
```

(continues on next page)

(continued from previous page)

```
-h, --help          show this help message and exit
-b BUILDSPEC, --buildspec BUILDSPEC
                    Specify path to buildspec (file, or directory) to validate
-x EXCLUDE, --exclude EXCLUDE
                    Specify path to buildspec to exclude (file or directory) during
↳ validation
-e EXECUTOR, --executor EXECUTOR
                    Specify buildspecs by executor name to validate
-t TAG, --tag TAG   Specify buildspecs by tag name to validate
```

The `-b` option can be used to specify path to buildspec file or directory to validate buildspecs. If its a directory, buildtest will traverse all directories recursively and find any `.yaml` file extensions and attempt to validate each buildspec. Shown below is an example output of what it may look like

```
$ buildtest buildspec validate -b tutorials/vars.yaml
Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/vars.yaml
All buildspecs passed validation!!!
```

If buildtest detects an error during validation, the error message will be displayed to screen as we see in this example

```
$ buildtest buildspec validate -b tutorials/invalid_tags.yaml

file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ tutorials/invalid_tags.yaml

-----
['network', 'network'] is not valid under any of the given schemas

Failed validating 'oneOf' in schema['properties']['tags']:
  {'oneOf': [{'type': 'string'},
              {'$ref': '#/definitions/list_of_strings'}]}

On instance['tags']:
  ['network', 'network']

Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/invalid_tags.yaml
There were 1 buildspecs that failed validation
```

Similarly we can search buildspecs based on tags if you want to validate a group of buildspecs using the `-t` option. We can append `-t` option multiple times to search by multiple tag names. In this next example, we will validate all buildspecs for **python** and **pass** tags.

```
$ buildtest buildspec validate -t python -t pass
Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/python-hello.yaml
Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/pass_returncode.yaml
Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/python-shell.yaml
All buildspecs passed validation!!!
```

Finally we can also search by executors using the `-e` option which can be appended to search by multiple executors.

```
$ buildtest buildspec validate -e generic.local.csh
Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/environment.yml
Processing buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/csh_shell_examples.yml
All buildspecs passed validation!!!
```

Edit buildspecs `buildtest edit`

Note: `buildtest et` is an alias for `buildtest edit` command.

The `buildtest edit` command can be used to edit buildspec with your preferred editor defined by environment `$EDITOR`, if this environment is not set `buildtest` will resort to `vim`. Once you make change, the file will be written back to disk and validated with the `jsonschema`. If it passes validation you will see a message such as follows:

```
$ buildtest edit tutorials/vars.yml
Writing file: /Users/siddiq90/Documents/GitHubDesktop/buildtest.tmp/tutorials/vars.yml
/Users/siddiq90/Documents/GitHubDesktop/buildtest.tmp/tutorials/vars.yml is valid
```

If there is an error during validation, `buildtest` will print the exception to stdout and it is your responsibility to fix the buildspec based on error message. In example below, the user provided an invalid value for `type` field.

```
$ buildtest edit tutorials/vars.yml
Writing file: /Users/siddiq90/Documents/GitHubDesktop/buildtest.tmp/tutorials/vars.yml
Traceback (most recent call last):
  File "/Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest", line 17, in
↳<module>
    buildtest.main.main()
  File "/Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/main.py", line 103,
↳in main
    edit_buildspec(args.buildspec, configuration)
  File "/Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/cli/edit.py", line
↳23, in edit_buildspec
    BuildspecParser(buildspec, be)
  File "/Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/buildsystem/parser.py
↳", line 74, in __init__
    self._validate()
  File "/Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/buildsystem/parser.py
↳", line 185, in _validate
    self._check_schema_type(test)
  File "/Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/buildsystem/parser.py
↳", line 101, in _check_schema_type
    raise BuildspecError(self.buildspec, msg)
buildtest.exceptions.BuildspecError: ' [/Users/siddiq90/Documents/GitHubDesktop/buildtest.
↳tmp/tutorials/vars.yml]: type script123 is not known to buildtest.'
```

Show buildspec buildtest buildspec show

buildtest can display content of buildspec file given a test name via `buildtest buildspec show` command which expects a positional argument that is the name of test. This can be quick way to see content of buildspec without remembering the full path to the buildspec.

In this next example, we will instruct buildtest to show content of buildspec for test name `python_hello`.

```
$ buildtest buildspec show python_hello
version: "1.0"
buildspecs:
  python_hello:
    type: script
    description: Hello World python
    executor: generic.local.bash
    tags: python
    run: python hello.py

buildspec: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ tutorials/python-hello.yml
```

There is bash completion for this command which will show list of test names available in the cache assuming you have run `buildtest buildspec find`. If you specify an invalid test name you will get an error followed by list of tests that are available in the cache

```
$ buildtest buildspec show XYZ123!
Traceback (most recent call last):
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
↳ buildtest", line 17, in <module>
    buildtest.main.main()
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/main.py", line 127, in main
    show_buildspecs(name=args.name, configuration=configuration)
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/cli/buildspec.py", line 950, in show_buildspecs
    f"{name} not in cache. Please select one of the following test: {cache.get_names()}"
buildtest.exceptions.BuildTestError: "XYZ123! not in cache. Please select one of the
↳ following test: ['skip', 'unskipped', 'status_regex_pass', 'status_regex_fail',
↳ 'metric_regex_example', 'executor_regex_script_schema', 'add_numbers', 'python_hello',
↳ '_bin_sh_shell', '_bin_bash_shell', 'bash_shell', 'sh_shell', 'shell_options', 'bash_
↳ login_shebang', 'bash_nonlogin_shebang', 'run_only_macos_distro', 'run_only_linux_
↳ distro', 'metric_variable_assignment', 'selinux_disable', 'exit1_fail', 'exit1_pass',
↳ 'returncode_list_mismatch', 'returncode_int_match', 'sleep', 'csh_shell', 'string_tag',
↳ 'list_of_strings_tags', 'bash_env_variables', 'csh_env_declaration', 'tcsh_env_
↳ declaration', 'hello_world', 'foo_bar', 'variables_bash', 'run_only_platform_darwin',
↳ 'run_only_platform_linux', 'circle_area', 'timelimit_min_max', 'timelimit_min',
↳ 'timelimit_max', 'timelimit_min_fail', 'timelimit_max_fail', 'run_only_as_root',
↳ 'spack_test', 'spack_env_directory', 'spack_sbatch_multi_executors', 'run_pre_post_
↳ commands', 'remove_environment_automatically', 'remove_environment_explicit', 'spack_
↳ test_results_specs_format', 'concretize_m4_in_spack_env', 'spack_env_create_from_
↳ manifest', 'spack_sbatch_example', 'add_mirror', 'add_mirror_in_spack_env', 'install_
↳ zlib', 'executors_sbatch_declaration', 'status_returncode_by_executors', 'executors_
↳ vars_env_declaration', 'openmp_hello_c_example', 'default_status_regex', 'override
↳ status_regex', 'override_environmentvars', 'custom_run_by_compilers', 'vecadd_gnu_
↳ exclude', 'hello_c', 'vecadd_gnu', 'hello_f', 'pre_post_build_run', 'metrics_variable_
↳ compiler', 'show_lsf_user_groups', 'show_host_groups', 'show_lsf_queues', 'show_lsf_queues_
↳ queues_formatted', 'show_lsf_queues_current_user', 'show_lsf_configuration', 'show_lsf_
↳ models', 'show_lsf_resources', 'lsf_version', 'display_lsf_hosts', 'display_hosts_
↳ format', 'bhosts_version', 'current_user_queue', 'show_all_jobs', 'show_accounts',
↳ (continues on next page)
```

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

```
usage: buildtest [options] [COMMANDS] report [-h] [--filter FILTER] [--format FORMAT] [--helpfilter] [--helpformat]
                                         [--latest] [--oldest] [-n] [-r REPORT] [-t]
                                         ...
```

[illegible]

```

-h, --help            show this help message and exit
--filter FILTER        Filter report by filter fields. The filter fields must be a
→key=value pair and multiple fields
                        can be comma separated in the following format: --filter_
→key1=val1,key2=val2 . For list of
                        filter fields run: --helpfilter.
--format FORMAT        format field for printing purposes. For more details see --
→helpformat for list of available
                        fields. Fields must be separated by comma (usage: --format
→<field1>,<field2>,...)
--helpfilter           List available filter fields to be used with --filter option
--helpformat           List of available format fields
--latest              Retrieve latest record of particular test
--oldest              Retrieve oldest record of particular test
-n, --no-header        Don't print headers column used with terse option (--terse).
-r REPORT, --report REPORT
                        Specify a report file to read
-t, --terse           Print output in machine readable format

```

SubComments:

```
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/report.json
```

(continues on next page)

(continued from previous page)

name	id	state	returncode	starttime	endtime	runtime	tags	buildspec
variables_bash	98b07db8	PASS	0	2021/09/09 15:54:50	2021/09/09 15:54:50	0.012987	tutorials	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml
variables_bash	7ff48178	PASS	0	2021/09/09 15:55:13	2021/09/09 15:55:13	0.042796	tutorials	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml
exit1_fail	53c2663d	FAIL	1	2021/09/09 15:54:51	2021/09/09 15:54:51	0.023217	tutorials fail	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml
exit1_fail	aaa388f4	FAIL	1	2021/09/09 15:55:18	2021/09/09 15:55:18	0.029501	tutorials fail	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml
exit1_fail	46c30f6b	FAIL	1	2021/09/09 15:55:18	2021/09/09 15:55:18	0.037522	tutorials fail	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml
exit1_fail	0de2fede	FAIL	1	2021/09/09 15:55:22	2021/09/09 15:55:22	0.020173	tutorials fail	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml

(continues on next page)

(continued from previous page)

```

| exit1_fail          | 69a9a391 | FAIL    |          1 | 2021/09/09
↪15:55:22 | 2021/09/09 15:55:22 | 0.017334 | tutorials fail    | /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml ↪
↪
+-----+-----+-----+-----+-----+
↪+-----+-----+-----+-----+-----+
↪+-----+-----+-----+-----+-----+
↪+-----+-----+-----+-----+-----+
| exit1_pass          | 97c4d576 | PASS    |          1 | 2021/09/09
↪15:54:51 | 2021/09/09 15:54:51 | 0.020955 | tutorials pass    | /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_returncode.yml ↪
↪
...

```

Note: `buildtest rt` is an alias for `buildtest report` command.

Format Reports (`buildtest report --format`)

Available Format Fields (`buildtest report --helpformat`)

The **buildtest report** command displays a default format fields that can be changed using the `--format` option. The report file (JSON) contains many more fields and we expose some of the fields with the `-format` option. To see a list of available format fields you can run `buildtest report --helpformat`. This option will list all format fields with their description.

```

$ buildtest report --helpformat
Fields      Description
-----
buildspec   Builds spec file
command     Command executed
compiler    Retrieve compiler used for test (applicable for compiler schema)
endtime     End Time for Test in date format
errfile     Error File
executor    Executor name
hostname    Retrieve hostname of machine where job was submitted from
full_id     Full qualified unique build identifier
id          Unique Build Identifier (abbreviated)
metrics     List all metrics if applicable
name        Name of test defined in buildspec
outfile     Output file
returncode  Return Code from Test Execution
runtime     Total runtime in seconds
schemafile  Schema file used for validation
starttime   Start Time of test in date format
state       Test State reported by buildtest (PASS/FAIL)
tags        Tag name
testroot    Root of test directory
testpath    Path to test
user        Get user who submitted job

```

Format Field Usage

The `--format` field are specified in comma separated format (i.e `--format <field1>,<field2>`). In this example we format table by fields `--format id,executor,state,returncode`.

```
$ buildtest rt --format name,id,executor,state,returncode
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/report.json

+-----+-----+-----+-----+-----+
↳ ---+
| name                | id      | executor          | state  |  ↳  |
↳ returncode |
+=====+=====+=====+=====+=====+
| variables_bash      | 98b07db8 | generic.local.bash | PASS   |  ↳  |
↳ 0 |
+-----+-----+-----+-----+-----+
↳ ---+
| variables_bash      | 7ff48178 | generic.local.bash | PASS   |  ↳  |
↳ 0 |
+-----+-----+-----+-----+-----+
↳ ---+
| exit1_fail          | 53c2663d | generic.local.sh   | FAIL   |  ↳  |
↳ 1 |
+-----+-----+-----+-----+-----+
↳ ---+
| exit1_fail          | aaa388f4 | generic.local.sh   | FAIL   |  ↳  |
↳ 1 |
+-----+-----+-----+-----+-----+
↳ ---+
| exit1_fail          | 46c30f6b | generic.local.sh   | FAIL   |  ↳  |
↳ 1 |
+-----+-----+-----+-----+-----+
↳ ---+
| exit1_fail          | 0de2fede | generic.local.sh   | FAIL   |  ↳  |
↳ 1 |
+-----+-----+-----+-----+-----+
↳ ---+
| exit1_fail          | 69a9a391 | generic.local.sh   | FAIL   |  ↳  |
↳ 1 |
+-----+-----+-----+-----+-----+
↳ ---+
| exit1_pass          | 97c4d576 | generic.local.sh   | PASS   |  ↳  |
↳ 1 |
+-----+-----+-----+-----+-----+
↳ ---+
...

```


Filter Reports (buildtest report --filter)

The **buildtest report** command will display all tests results, which can be quite long depending on number of tests so therefore we need a mechanism to filter the test results. The **--filter** option can be used to filter out tests in the output based on filter fields. First, lets see the available filter fields by run **buildtest report --helpfilter** which shows a list of filter fields and their description.

```
$ buildtest report --helpfilter
```

Filter Fields	Description	Expected Value
-----	-----	-----
buildspec	Filter by buildspec file	FILE
name	Filter by test name	STRING
executor	Filter by executor name	STRING
state	Filter by test state	PASS/FAIL
tags	Filter tests by tag name	STRING
returncode	Filter tests by returncode	INT

The **--filter** option expects arguments in **key=value** format. You can specify multiple filter delimited by comma. buildtest will treat multiple filters as logical **AND** operation. The filter option can be used with **--format** field. Let's see some examples to illustrate the point.

Filter by returncode (--filter returncode)

If you want to retrieve all tests with a given returncode, we can use the **returncode** property. For instance, let's retrieve all tests with returncode of 2 by setting **--filter returncode=2**.

```
$ buildtest rt --filter returncode=2 --format=name,id,returncode
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/report.json
```

name	id	returncode
returncode_list_mismatch	31d04ee2	2
returncode_list_mismatch	5e2d005d	2
returncode_list_mismatch	1f4bb7da	2
returncode_list_mismatch	3f3f660f	2
returncode_list_mismatch	866fd8e4	2

Note: buildtest automatically converts returncode to integer when matching returncode, so **--filter returncode="2"** will work too

Filter by test name (`--filter name`)

If you want to filter by test name, use the **name** attribute in filter option. Let's assume we want to filter all tests by name `exit1_pass`, this can be achieved by setting filter field as follows: `--filter name=exit1_pass`. Shown below is an example using **name** filter field to filter test results.

```
$ buildtest rt --filter name=exit1_pass --format=name,id,returncode,state
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/report.json
```

name	id	returncode	state
exit1_pass	97c4d576	1	PASS
exit1_pass	55601e7e	1	PASS
exit1_pass	b2c32941	1	PASS
exit1_pass	80376cc8	1	PASS
exit1_pass	bdda64f7	1	PASS
exit1_pass	6d8369a3	1	PASS

Filter by buildspec (`--filter buildspec`)

Likewise, we can filter results by buildspec file using **buildspec** attribute via `--filter buildspec=<file>`. The **buildspec** attribute must resolve to a file path which can be relative or absolute path. buildtest will resolve path (absolute path) and find the appropriate tests that belong to the buildspec file. If file doesn't exist or is not found in cache it will raise an error.

```
$ buildtest rt --filter buildspec=tutorials/python-hello.yml --format=name,id,state,
↳buildspec
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/report.json
```

name	id	state	buildspec
python_hello	6feaf262	PASS	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-hello.yml
python_hello	3c422a2b	PASS	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-hello.yml

Filter by test state (`--filter state`)

If you want to filter results by test state, use the **state** property. This can be useful if you want to know all pass or failed tests. The state property expects value of [PASS|FAIL] since these are the two recorded test states marked by buildtest. We can also pass multiple filter fields for instance if we want to find all **FAIL** tests for executor **generic.local.sh** we can do the following.

```
$ buildtest rt --filter state=FAIL,executor=generic.local.sh --format=name,id,state,
↳ executor
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/report.json
```

name	id	state	executor
exit1_fail	53c2663d	FAIL	generic.local.sh
exit1_fail	aaa388f4	FAIL	generic.local.sh
exit1_fail	46c30f6b	FAIL	generic.local.sh
exit1_fail	0de2fede	FAIL	generic.local.sh
exit1_fail	69a9a391	FAIL	generic.local.sh
returncode_list_mismatch	31d04ee2	FAIL	generic.local.sh
returncode_list_mismatch	5e2d005d	FAIL	generic.local.sh
returncode_list_mismatch	1f4bb7da	FAIL	generic.local.sh
returncode_list_mismatch	3f3f660f	FAIL	generic.local.sh
returncode_list_mismatch	866fd8e4	FAIL	generic.local.sh
timelimit_min_fail	5352acc1	FAIL	generic.local.sh
timelimit_max_fail	40bee7c5	FAIL	generic.local.sh
status_returncode_by_executors	912c478d	FAIL	generic.local.sh

Filter Exception Cases

The returncode filter field expects an integer value, so if you try a non-integer returncode you will get the following message

```
$ buildtest rt --filter returncode=1.5
Traceback (most recent call last):
  File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
↳ buildtest", line 17, in <module>
    buildtest.main.main()
```

(continues on next page)

(continued from previous page)

```
File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/main.py", line 151, in main
    report_cmd(args)
File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/cli/report.py", line 670, in report_cmd
    report_file=args.report,
File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/cli/report.py", line 96, in __init__
    self._check_filter_fields()
File "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/cli/report.py", line 133, in _check_filter_fields
    f"Invalid returncode:{self.filter[key]} must be an integer"
buildtest.exceptions.BuildTestError: 'Invalid returncode:1.5 must be an integer'
```

The state filter field expects value of PASS or FAIL so if you specify an invalid state you will get an error as follows.

```
$ buildtest rt --filter state=UNKNOWN
filter argument 'state' must be 'PASS' or 'FAIL' got value UNKNOWN
```

The buildspec field expects a valid file path, it can be an absolute or relative path, buildtest will resolve absolute path and check if file exist and is in the report file. If it's an invalid file we get an error such as

```
$ buildtest rt --filter buildspec=/path/to/invalid.yml
Invalid File Path for filter field 'buildspec': /path/to/invalid.yml
```

You may have a valid filepath for buildspec filter field such as \$BUILDTEST_ROOT/tutorials/invalid_executor.yml, but there is no record of a test in the report cache because this test wasn't run. In this case you will get the following message.

```
$ buildtest rt --filter buildspec=$BUILDTEST_ROOT/tutorials/invalid_executor.yml
buildspec file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/tutorials/invalid_executor.yml not found in cache
```

Find Latest or Oldest test

We can search for oldest or latest test for any given test. This can be useful if you want to see first or last test run. If you want to retrieve the oldest test you can use --oldest option. buildtest will append tests, therefore last record in dictionary will be latest record, similarly first record is the oldest record.

Let's take a look at this example, we filter by test name hello_f which retrieves three entries. Now let's filter by oldest record by specifying --oldest option and it will retrieve the first record which is test id **349f3ada**.

```
$ buildtest report --filter name=hello_f --format name,id,starttime
Reading Report File: /Users/siddiq90/.buildtest/report.json

+-----+-----+-----+
| name   | id      | starttime |
+=====+=====+=====+
| hello_f | 349f3ada | 2021/02/11 18:13:08 |
+-----+-----+-----+
| hello_f | ecd4a3f2 | 2021/02/11 18:13:18 |
+-----+-----+-----+
```

(continues on next page)

(continued from previous page)

```
| hello_f | 5c87978b | 2021/02/11 18:13:33 |
+-----+-----+-----+

$ buildtest report --filter name=hello_f --format name,id,starttime --oldest
Reading Report File: /Users/siddiq90/.buildtest/report.json

+-----+-----+-----+
| name   | id      | starttime          |
+=====+=====+=====+
| hello_f | 349f3ada | 2021/02/11 18:13:08 |
+-----+-----+-----+
```

If you want to retrieve the latest test result you can use `--latest` option which will retrieve the last record, in the same example we will retrieve test id `5c87978b`.

```
$ buildtest report --filter name=hello_f --format name,id,starttime --latest
Reading Report File: /Users/siddiq90/.buildtest/report.json

+-----+-----+-----+
| name   | id      | starttime          |
+=====+=====+=====+
| hello_f | 5c87978b | 2021/02/11 18:13:33 |
+-----+-----+-----+
```

You may combine `--oldest` and `--latest` options in same command, in this case buildtest will retrieve the first and last record of every test.

```
$ buildtest report --format name,id,starttime --oldest --latest | more
Reading Report File: /Users/siddiq90/.buildtest/report.json

+-----+-----+-----+
| name           | id      | starttime          |
+=====+=====+=====+
| variables_bash | 750f48bc | 2021/02/11 18:13:03 |
+-----+-----+-----+
| variables_bash | 1bdfd403 | 2021/02/11 18:13:32 |
+-----+-----+-----+
| ulimit_filelock_unlimited | b7b852e4 | 2021/02/11 18:13:03 |
+-----+-----+-----+
| ulimit_filelock_unlimited | 56345a43 | 2021/02/11 18:13:18 |
+-----+-----+-----+
```

Terse Output

If you would like to parse the result of `buildtest report`, you can use the `--terse` or `-t` option which will print the report in machine readable format that shows the name of each column followed by each entry. Each entry is delimited by PIPE symbol (`|`). The `--terse` option works with `--format` and `--filter` option. In this next example, we report all FAIL tests in terse output. The first line is the header of tables followed by output, if you want to disable output of header you can use `--no-header` option.

```
$ buildtest report --filter state=FAIL --format=name,id,state -t
name|id|state
exit1_fail|53c2663d|FAIL
exit1_fail|aaa388f4|FAIL
exit1_fail|46c30f6b|FAIL
exit1_fail|0de2fedf|FAIL
exit1_fail|69a9a391|FAIL
returncode_list_mismatch|31d04ee2|FAIL
returncode_list_mismatch|5e2d005d|FAIL
returncode_list_mismatch|1f4bb7da|FAIL
returncode_list_mismatch|3f3f660f|FAIL
returncode_list_mismatch|866fd8e4|FAIL
status_regex_fail|099fdf40|FAIL
timelimit_min_fail|5352acc1|FAIL
timelimit_max_fail|40bee7c5|FAIL
status_returncode_by_executors|912c478d|FAIL
ulimit_stacksize_unlimited|7204f37a|FAIL
ulimit_stacksize_unlimited|f2782a7c|FAIL
ulimit_filedescriptor_4096|677a3ea5|FAIL
ulimit_filedescriptor_4096|fe1869b1|FAIL
ulimit_max_user_process_2048|81105021|FAIL
ulimit_max_user_process_2048|7d195a72|FAIL
kernel_swapusage|c6406429|FAIL
kernel_swapusage|5ef212a6|FAIL
kernel_swapusage|92b2bafa|FAIL
systemd_default_target|daaa00e7|FAIL
systemd_default_target|d016e48c|FAIL
systemd_default_target|e16f7e53|FAIL
ssh_localhost_remotecommand|d99f5389|FAIL
ssh_localhost_remotecommand|93086006|FAIL
ssh_localhost_remotecommand|c07d8422|FAIL
```

Report Summary (buildtest report summary)

The `buildtest report summary` command can be used to provide a summary of the test report with breakdown statistics of tests including all fail tests, number of tests by name, test runs and buildsspecs in report file.

Shown below is an example output from the report summary.

```
$ buildtest report summary
Report: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ var/report.json
Total Tests: 79
Total Tests by Names: 37
Number of buildsspecs in report: 22
```

Breakdown by Test

name	runs	pass	fail
variables_bash	2	2	0

(continues on next page)

(continued from previous page)

exit1_fail		5		0		5	
exit1_pass		6		6		0	
returncode_list_mismatch		5		0		5	
returncode_int_match		6		6		0	
status_regex_pass		1		1		0	
status_regex_fail		1		0		1	
timelimit_min_max		1		1		0	
timelimit_min		1		1		0	
timelimit_max		1		1		0	
timelimit_min_fail		1		0		1	
timelimit_max_fail		1		0		1	
bash_login_shebang		1		1		0	
bash_nonlogin_shebang		1		1		0	
unskipped		1		1		0	
metric_regex_example		1		1		0	
metric_variable_assignment		1		1		0	
executor_regex_script_schema		2		2		0	
executors_vars_env_declaration		2		2		0	
executors_sbatch_declaration		2		2		0	
status_returncode_by_executors		2		1		1	
ulimit_filelock_unlimited		2		2		0	
ulimit_cputime_unlimited		2		2		0	
ulimit_stacksize_unlimited		2		0		2	
ulimit_vmsize_unlimited		2		2		0	
ulimit_filedescriptor_4096		2		0		2	
ulimit_max_user_process_2048		2		0		2	

(continues on next page)

(continued from previous page)

kernel_swapusage	3	0	3
root_disk_usage	3	3	0
systemd_default_target	3	0	3
ssh_localhost_remotecommand	3	0	3
string_tag	1	1	0
list_of_strings_tags	1	1	0
python_hello	2	2	0
circle_area	5	5	0
run_only_platform_linux	1	1	0
hello_world	1	1	0

FAIL test

name	id	executor	state	
returncode runtime				
exit1_fail	53c2663d	generic.local.sh	FAIL	
1 0.023217				
exit1_fail	aaa388f4	generic.local.sh	FAIL	
1 0.029501				
exit1_fail	46c30f6b	generic.local.sh	FAIL	
1 0.037522				
exit1_fail	0de2fede	generic.local.sh	FAIL	
1 0.020173				
exit1_fail	69a9a391	generic.local.sh	FAIL	
1 0.017334				
returncode_list_mismatch	31d04ee2	generic.local.sh	FAIL	
2 0.019587				

(continues on next page)

(continued from previous page)

returncode_list_mismatch	5e2d005d	generic.local.sh	FAIL		
↪2 0.018428					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
returncode_list_mismatch	1f4bb7da	generic.local.sh	FAIL		
↪2 0.017392					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
returncode_list_mismatch	3f3f660f	generic.local.sh	FAIL		
↪2 0.019207					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
returncode_list_mismatch	866fd8e4	generic.local.sh	FAIL		
↪2 0.017478					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
status_regex_fail	099fdf40	generic.local.bash	FAIL		
↪0 0.025702					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
timelimit_min_fail	5352acc1	generic.local.sh	FAIL		
↪0 2.03991					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
timelimit_max_fail	40bee7c5	generic.local.sh	FAIL		
↪0 3.03514					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
status_returncode_by_executors	912c478d	generic.local.sh	FAIL		
↪0 0.013939					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
ulimit_stacksize_unlimited	7204f37a	generic.local.bash	FAIL		
↪0 0.023393					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
ulimit_stacksize_unlimited	f2782a7c	generic.local.bash	FAIL		
↪0 0.023499					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
ulimit_filedescriptor_4096	677a3ea5	generic.local.bash	FAIL		
↪0 0.008116					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
ulimit_filedescriptor_4096	fe1869b1	generic.local.bash	FAIL		
↪0 0.027482					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					
ulimit_max_user_process_2048	81105021	generic.local.bash	FAIL		
↪0 0.021057					
+-----+-----+-----+-----+-----+-----+					
↪+-----+					

(continues on next page)

(continued from previous page)

ulimit_max_user_process_2048	7d195a72	generic.local.bash	FAIL		└
↪0 0.02491					
+-----+-----+-----+-----+-----+					
↪+-----+					
kernel_swapusage	c6406429	generic.local.bash	FAIL		└
↪127 0.020363					
+-----+-----+-----+-----+-----+					
↪+-----+					
kernel_swapusage	5ef212a6	generic.local.bash	FAIL		└
↪127 0.031774					
+-----+-----+-----+-----+-----+					
↪+-----+					
kernel_swapusage	92b2bafa	generic.local.bash	FAIL		└
↪127 0.027239					
+-----+-----+-----+-----+-----+					
↪+-----+					
systemd_default_target	daaa00e7	generic.local.bash	FAIL		└
↪1 0.040417					
+-----+-----+-----+-----+-----+					
↪+-----+					
systemd_default_target	d016e48c	generic.local.bash	FAIL		└
↪1 0.030873					
+-----+-----+-----+-----+-----+					
↪+-----+					
systemd_default_target	e16f7e53	generic.local.bash	FAIL		└
↪1 0.027696					
+-----+-----+-----+-----+-----+					
↪+-----+					
ssh_localhost_remotecommand	d99f5389	generic.local.bash	FAIL		└
↪255 0.155613					
+-----+-----+-----+-----+-----+					
↪+-----+					
ssh_localhost_remotecommand	93086006	generic.local.bash	FAIL		└
↪255 0.032854					
+-----+-----+-----+-----+-----+					
↪+-----+					
ssh_localhost_remotecommand	c07d8422	generic.local.bash	FAIL		└
↪255 0.030714					
+-----+-----+-----+-----+-----+					
↪+-----+					

Inspect Tests Records via buildtest inspect

Note: buildtest it is an alias for buildtest inspect command.

In previous examples we saw how we can retrieve test records using `buildtest report` which is printed in table format. We have limited the output to a limited fields however, if you want to analyze a particular, we have a separate command called `buildtest inspect` that can be used for inspecting a test record based on name or id. Shown below is the command usage for *buildtest inspect* command.

```
$ buildtest inspect --help
usage: buildtest [options] [COMMANDS] inspect [-h] [-r REPORT] ...

optional arguments:
  -h, --help            show this help message and exit
  -r REPORT, --report REPORT
                        Specify a report file to load when inspecting test

subcommands:
  Inspect Test result based on Test ID or Test Name

  buildspec            Inspect a test based on buildspec
  id                   Specify a Test ID
  name                 Specify name of test
  query               Query fields from record
  list                 List all test names, ids, and corresponding buildspecs
```

You can report all test names and corresponding ids using `buildtest inspect list` which will be used for querying tests by name or id.

```
$ buildtest inspect list
+-----+-----+-----+
| name | id | buildspec |
+-----+-----+-----+
| variables_bash | 98b07db8-9b3e-4ff0-b526-1ebcebbc3a1e | /home/docs/
| checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml |
+-----+-----+-----+
| variables_bash | 7ff48178-2f82-4dc0-8e9b-17ce6ff0b48f | /home/docs/
| checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml |
+-----+-----+-----+
| exit1_fail | 53c2663d-6290-4715-bb97-eb2f8b99ff86 | /home/docs/
| checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
| returncode.yml |
+-----+-----+-----+
| exit1_fail | aaa388f4-6e71-4e73-aabd-4f5fb8e317ad | /home/docs/
| checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
| returncode.yml |
+-----+-----+-----+
```

(continues on next page)

(continued from previous page)

```
| exit1_fail | 46c30f6b-5121-4715-90b8-961091a04665 | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
↪returncode.yml |
+-----+-----+
↪
↪
| exit1_fail | 0de2fede-81b4-4f55-83d2-4a85e73f96b6 | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
↪returncode.yml |
+-----+-----+
↪
↪
| exit1_fail | 69a9a391-cdce-4269-adb5-b844fb4ad23a | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
↪returncode.yml |
+-----+-----+
↪
↪
| exit1_pass | 97c4d576-0e99-4f02-abdc-aa8b54802834 | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
↪returncode.yml |
+-----+-----+
↪
↪
| exit1_pass | 55601e7e-a451-4fc0-81bb-5138115297a9 | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/pass_
↪returncode.yml |
...
```

Inspecting Test by Name via `buildtest inspect name`

The `buildtest inspect name` expects a list of positional argument that correspond to name of test you want to query and `buildtest` will fetch the **last** record for each named test. Let's see an example to illustrate the point. We can see that each test is stored as a JSON format and `buildtest` keeps track of metadata for each test such as *user*, *hostname*, *command*, path to output and error file, content of test, state of test, returncode, etc... In this example, we will retrieve record for test name **circle_area** which will print the raw content of the test in JSON format.

```
$ buildtest it name circle_area
{
  "circle_area": {
    "id": "8e5d3510",
    "full_id": "8e5d3510-bbe7-4140-86d7-aa168c6babec",
    "description": "Calculate circle of area given a radius",
    "schemafile": "script-v1.0.schema.json",
    "executor": "generic.local.python",
    "compiler": null,
    "hostname": "build-14673784-project-280831-buildtest",
    "user": "docs",
    "testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510",
    "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510/circle_area.sh"
  }
}
```

(continued from previous page)

```

"stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510/stage",
"command": "sh circle_area_build.sh",
"outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510/circle_area.out",
"errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510/circle_area.err",
"buildspec_content": "version: \"1.0\"\\nbuilder:\\n circle_area:\\n executor:\\n
↪generic.local.python\\n type: script\\n shell: python\\n description: \\n
↪\"Calculate circle of area given a radius\"\\n tags: [tutorials, python]\\n run: |\\n
↪import math\\n radius = 2\\n area = math.pi * radius * radius\\n
↪print(\"Circle Radius \", radius)\\n print(\"Area of circle \", area)\\n\",
"test_content": "#!/bin/bash\\npython /home/docs/checkouts/readthedocs.org/user_
↪builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↪area/8e5d3510/stage/circle_area.py",
"buildscript_content": "#!/bin/bash\\n\\n\\n##### START VARIABLE DECLARATION ###
↪#####\\nexport BUILDTEST_TEST_NAME=circle_area\\nexport BUILDTEST_TEST_
↪ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪tests/generic.local.python/python-shell/circle_area/8e5d3510\\nexport BUILDTEST_
↪BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/tutorials\\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/readthedocs.org/user_
↪builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↪area/8e5d3510/stage\\nexport BUILDTEST_TEST_ID=8e5d3510-bbe7-4140-86d7-aa168c6babec\\n###
↪##### END VARIABLE DECLARATION #####\\n\\n\\n# source executor_
↪startup script\\nsource /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/executor/generic.local.python/before_script.sh\\n# Run generated_
↪script\\n/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪var/tests/generic.local.python/python-shell/circle_area/8e5d3510/stage/circle_area.sh\\n
↪# Get return code\\nreturncode=$?\\n# Exit with return code\\nexit $returncode",
"logpath": "/tmp/buildtest_z7y7g10g.log",
"metrics": {},
"check": {
"returncode": "N/A",
"regex": "N/A",
"runtime": "N/A"
},
"tags": "tutorials python",
"starttime": "2021/09/09 15:55:22",
"endtime": "2021/09/09 15:55:22",
"runtime": "0.111269",
"state": "PASS",
"returncode": "0",
"output": "Circle Radius 2\\nArea of circle 12.566370614359172\\n",
"error": "circle_area_build.sh: 14: circle_area_build.sh: source: not found\\n",
"job": {},
"build_script": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_area/8e5d3510/
↪circle_area_build.sh"
}
}

```

You can query multiple tests as positional arguments in the format: `buildtest inspect name <test1> <test2>`
 In this next example, we will retrieve test records for `bash_shell` and `python_hello`.

```
$ buildtest inspect name bash_shell python_hello
{
  "python_hello": {
    "id": "3c422a2b",
    "full_id": "3c422a2b-81b4-4d4f-9240-1e74655d9c8e",
    "description": "Hello World python",
    "schemafile": "script-v1.0.schema.json",
    "executor": "generic.local.bash",
    "compiler": null,
    "hostname": "build-14673784-project-280831-buildtest",
    "user": "docs",
    "testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b",
    "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/python_hello.sh",
    "stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/stage",
    "command": "sh python_hello_build.sh",
    "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/python_hello.out",
    "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/python_hello.err",
    "buildspec_content": "version: \"1.0\"\\nbuildspecs:\\n  python_hello:\\n    type:↪
↪ script\\n    description: Hello World python\\n    executor: generic.local.bash\\n    ↪
↪ tags: python\\n    run: python hello.py\\n\\n",
    "test_content": "#!/bin/bash \\n# Content of run section\\npython hello.py",
    "buildscript_content": "#!/bin/bash\\n\\n##### START VARIABLE DECLARATION ###
↪ #####\\nexport BUILDTEST_TEST_NAME=python_hello\\nexport BUILDTEST_TEST_
↪ ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↪ tests/generic.local.bash/python-hello/python_hello/3c422a2b\\nexport BUILDTEST_
↪ BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/tutorials\\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/readthedocs.org/user_
↪ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
↪ hello/3c422a2b/stage\\nexport BUILDTEST_TEST_ID=3c422a2b-81b4-4d4f-9240-1e74655d9c8e\\n##
↪ ##### END VARIABLE DECLARATION #####\\n\\n\\n# source executor↪
↪ startup script\\nsource /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪ checkouts/v0.11.0/var/executor/generic.local.bash/before_script.sh\\n# Run generated↪
↪ script\\n/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪ var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/stage/python_hello.sh\\n
↪ # Get return code\\nreturncode=$?\\n# Exit with return code\\nexit $returncode",
    "logpath": "/tmp/buildtest_tg3mt2bz.log",
    "metrics": {},
    "check": {
      "returncode": "N/A",
      "regex": "N/A",
      "runtime": "N/A"
    },
    "tags": "python",
    "starttime": "2021/09/09 15:55:18",
    "endtime": "2021/09/09 15:55:18",
    "runtime": "0.092008",
    "state": "PASS",
    "returncode": "0",
```

(continues on next page)

(continued from previous page)

```

    "output": "Hello World\n",
    "error": "python_hello_build.sh: 14: python_hello_build.sh: source: not found\n",
    "job": {},
    "build_script": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_hello/3c422a2b/
↪python_hello_build.sh"
  }
}

```

If you want to query all test records for a given name you can use the `--all` option which is applied to all positional arguments.

Inspect Test by buildspec via `buildtest inspect buildspec`

buildtest can fetch records based on buildspec via `buildtest inspect buildspec` which expects a list of buildspecs. By default, buildtest will fetch the latest record of each test, but if you want to fetch all records you can pass the `--all` option.

In example below we will fetch latest record for all tests in `tutorials/vars.yml`

```

$ buildtest it buildspec tutorials/vars.yml
{
  "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪tutorials/vars.yml": {
    "variables_bash": {
      "id": "7ff48178",
      "full_id": "7ff48178-2f82-4dc0-8e9b-17ce6ff0b48f",
      "description": "Declare shell variables in bash",
      "schemafile": "script-v1.0.schema.json",
      "executor": "generic.local.bash",
      "compiler": null,
      "hostname": "build-14673784-project-280831-buildtest",
      "user": "docs",
      "testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178",
      "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash.sh",
      "stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/stage",
      "command": "sh variables_bash_build.sh",
      "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash.out",
      "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash.err",
      "buildspec_content": "version: \"1.0\"\nbuildspecs:\n  variables_bash:\n    type:
↪script\n    executor: generic.local.bash\n    description: Declare shell variables in
↪bash\n    tags: [tutorials]\n    vars:\n      X: 1\n      Y: 2\n      literalstring: |\n
↪  \"this is a literal string ':' '\n      singlequote: \"'singlequote'\n
↪  doublequote: \"\"doublequote\"\"\n      current_user: \"$(whoami)\"\n      num_files:
↪  \"`find $HOME -type f -maxdepth 1 | wc -l`\"\n      run: |\n        echo \"$X+$Y=$((($X+
↪$Y))\n        echo $literalstring\n        echo $singlequote\n        echo $doublequote\n
↪        echo \"current user:\" $current_user\n        echo \"number of files:\" $num_files",

```

(continues on next page)

(continued from previous page)

```
"test_content": "#!/bin/bash\n# Declare shell variables\nX=1\nY=2\nliteralstring=\nthis is a literal string ':' '\n\nsinglequote='\nsinglequote'\nsdoublequote='\n\ndoublequote'\nncurrent_user=$(whoami)\nnnum_files=`find $HOME -type f -maxdepth 1 |wc -l`\n\nContent of run section\nnecho \"$X+$Y=$((($X+$Y))\nnecho $literalstring\nnecho $singlequote\nnecho $doublequote\nnecho \"current user:\" $current_user\nnecho \nnumber of files:\" $num_files\",  
\"buildscript_content\": \"#!/bin/bash\n\n##### START VARIABLE DECLARATION  
#####\nexport BUILDTEST_TEST_NAME=variables_bash\nexport BUILDTEST_TEST_ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178\nexport BUILDTEST_BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/stage\nexport BUILDTEST_TEST_ID=7ff48178-2f82-4dc0-8e9b-17ce6ff0b48f\n#####\nEND VARIABLE DECLARATION #####\n\nsource executor startup_script\nsource /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/executor/generic.local.bash/before_script.sh\nRun generated script\n/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/stage/variables_bash.sh\nGet return code\nreturncode=$?\nExit with return code\nexit $returncode\",  
\"logpath\": \"/tmp/buildtest_hkdfz2sm.log\",  
\"metrics\": {},  
\"check\": {  
    \"returncode\": \"N/A\",  
    \"regex\": \"N/A\",  
    \"runtime\": \"N/A\"  
},  
\"tags\": \"tutorials\",  
\"starttime\": \"2021/09/09 15:55:13\",  
\"endtime\": \"2021/09/09 15:55:13\",  
\"runtime\": \"0.042796\",  
\"state\": \"PASS\",  
\"returncode\": \"0\",  
\"output\": \"1+2=3\nthis is a literal string ':'\n'singlequote'\n'doublequote'\nncurrent user: docs\nnumber of files: 4\n\",  
\"error\": \"variables_bash_build.sh: 14: variables_bash_build.sh: source: not found\n\",  
\"job\": {},  
\"build_script\": \"/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash_build.sh\"  
}  
}
```

buildtest will report an error if an input buildspec is invalid filepath such as one below

```
$ buildtest it buildspec /tmp/buildspec.yml
buildspec: /tmp/buildspec.yml is not valid file
There are no buildspects in cache based on input buildspects: ['/tmp/buildspec.yml']
```

You can also pass multiple buildspecs on the command line and fetch all records for a test. In example below we will fetch all records from buildspecs **tutorials/vars.yml** and **tutorials/status_regex.yml**


```

$ buildtest it buildspec --all tutorials/vars.yml tutorials/status_regex.yml
{
  "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
  ↪tutorials/vars.yml": {
    "variables_bash": [
      {
        "id": "98b07db8",
        "full_id": "98b07db8-9b3e-4ff0-b526-1ebcebbc3a1e",
        "description": "Declare shell variables in bash",
        "schemafile": "script-v1.0.schema.json",
        "executor": "generic.local.bash",
        "compiler": null,
        "hostname": "build-14673784-project-280831-buildtest",
        "user": "docs",
        "testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
        ↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8",
        "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
        ↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_
        ↪bash.sh",
        "stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
        ↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/stage",
        "command": "sh variables_bash_build.sh",
        "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
        ↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash.out",
        "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
        ↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash.err",
        "buildspec_content": "version: \"1.0\"\\nbuildspecs:\\n  variables_bash:\\n
        ↪type: script\\n    executor: generic.local.bash\\n    description: Declare shell
        ↪variables in bash\\n    tags: [tutorials]\\n    vars:\\n      X: 1\\n      Y: 2\\n
        ↪literalstring: |\\n      \\\"this is a literal string ':' '\\\"\\n      singlequote: \\
        ↪'singlequote\\\"\\\"\\n      doublequote: \\\"\\\"doublequote\\\"\\\"\\n      current_user: \\
        ↪$(whoami)\\\"\\n      num_files: \\\"find $HOME -type f -maxdepth 1 | wc -l\\\"\\n\\n      run:
        ↪|\\n      echo \\\"$X+$Y=\\\"$((($X+$Y))\\n      echo $literalstring\\n      echo $singlequote
        ↪\\n      echo $doublequote\\n      echo \\\"current user:\\\" $current_user\\n      echo \\
        ↪\"number of files:\\\" $num_files\",
        "test_content": "#!/bin/bash \\n# Declare shell variables\\nX=1\\nY=2\\
        ↪nliteralstring=\\\"this is a literal string ':' '\\\"\\n\\nsinglequote=\\\"\\\"singlequote\\\"\\
        ↪ndoublequote=\\\"\\\"doublequote\\\"\\\"\\n\\ncurrent_user=$(whoami)\\nnum_files=`find $HOME -type
        ↪f -maxdepth 1 | wc -l\\\"\\n\\n\\n# Content of run section\\necho \\\"$X+$Y=\\\"$((($X+$Y))\\necho
        ↪$literalstring\\necho $singlequote\\necho $doublequote\\necho \\\"current user:\\\" $current_
        ↪user\\necho \\\"number of files:\\\" $num_files\",
        "buildscript_content": "#!/bin/bash\\n\\n##### START VARIABLE
        ↪DECLARATION #####\\nexport BUILDTEST_TEST_NAME=variables_bash\\
        ↪nexport BUILDTEST_TEST_ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
        ↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8\\nexport
        ↪BUILDTEST_BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
        ↪checkouts/v0.11.0/tutorials\\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/
        ↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
        ↪vars/variables_bash/98b07db8/stage\\nexport BUILDTEST_TEST_ID=98b07db8-9b3e-4ff0-b526-
        ↪1ebcebbc3a1e\\n##### END VARIABLE DECLARATION #####\\n\\n\\n
        ↪source executor startup script\\nsource /home/docs/checkouts/readthedocs.org/user_
        ↪builds/buildtest/checkouts/v0.11.0/var/executor/generic.local.bash/before_script.sh\\n
        ↪Run generated script\\n/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
        ↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/stage/
        ↪variables_bash.sh\\n# Get return code\\nreturncode=$?\\n# Exit with return code\\nexit
        ↪$returncode"
      }
    ]
  }
}

```

(continued from previous page)

```

"logpath": "/tmp/buildtest_8xn_louq.log",
"metrics": {},
"check": {
  "returncode": "N/A",
  "regex": "N/A",
  "runtime": "N/A"
},
"tags": "tutorials",
"starttime": "2021/09/09 15:54:50",
"endtime": "2021/09/09 15:54:50",
"runtime": "0.012987",
"state": "PASS",
"returncode": "0",
"output": "1+2=3\nthis is a literal string ':'\n'singlequote'\n\"doublequote\"\\
↪ncurrent user: docs\nnumber of files: 4\n",
  "error": "variables_bash_build.sh: 14: variables_bash_build.sh: source: not_
↪found\n",
  "job": {},
  "build_script": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_
↪bash_build.sh"
},
{
  "id": "7ff48178",
  "full_id": "7ff48178-2f82-4dc0-8e9b-17ce6ff0b48f",
  "description": "Declare shell variables in bash",
  "schemafile": "script-v1.0.schema.json",
  "executor": "generic.local.bash",
  "compiler": null,
  "hostname": "build-14673784-project-280831-buildtest",
  "user": "docs",
  "testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178",
  "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_
↪bash.sh",
  "stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/stage",
  "command": "sh variables_bash_build.sh",
  "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash.out",
  "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_bash.err",
  "buildspec_content": "version: \"1.0\"\n\nbuildspecs:\n  variables_bash:\n    ↪
↪type: script\n    executor: generic.local.bash\n    description: Declare shell_
↪variables in bash\n    tags: [tutorials]\n    vars:\n      X: 1\n      Y: 2\n    ↪
↪literalstring: |\n      \"this is a literal string ':' \"\n      singlequote: \\
↪'singlequote\\'\n      doublequote: \\\"doublequote\\\"\\\"\\n      current_user: \"
↪$(whoami)\"\n      num_files: \"`find $HOME -type f -maxdepth 1 | wc -l`\"\n\n      run:
↪|\n        echo \"$X+$Y=$((X+Y))\"\n        echo $literalstring\n        echo $singlequote\
↪n        echo $doublequote\n        echo \"current user:\" $current_user\n        echo \
↪\"number of files:\" $num_files",

```

(continues on next page)

(continued from previous page)

```

    "test_content": "#!/bin/bash \n# Declare shell variables\nX=1\nY=2\
↪nliteralstring=\"this is a literal string ':' '\n\nsinglequote=\"'\singlequote\\'\
↪ndoublequote=\"\\\"doublequote\\\"\"\ncurrent_user=$(whoami)\nnum_files=`find $HOME -type\
↪f -maxdepth 1 | wc -l`\n\n\n# Content of run section\necho \"$X+$Y=\"${($X+$Y)}\necho
↪$literalstring\necho $singlequote\necho $doublequote\necho \"current user:\" $current_
↪user\necho \"number of files:\" $num_files",
    "buildscript_content": "#!/bin/bash\n\n\n##### START VARIABLE\
↪DECLARATION #####\n\nexport BUILDTEST_TEST_NAME=variables_bash\
↪nexport BUILDTEST_TEST_ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178\nexport\
↪BUILDTEST_BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/tutorials\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↪vars/variables_bash/7ff48178/stage\nexport BUILDTEST_TEST_ID=7ff48178-2f82-4dc0-8e9b-
↪17ce6ff0b48f\n##### END VARIABLE DECLARATION #####\n\n\n#\
↪source executor startup script\nsource /home/docs/checkouts/readthedocs.org/user_
↪builds/buildtest/checkouts/v0.11.0/var/executor/generic.local.bash/before_script.sh\n#\
↪Run generated script\n/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/stage/
↪variables_bash.sh\n# Get return code\nreturncode=$?\n# Exit with return code\nexit
↪$returncode",
    "logpath": "/tmp/buildtest_hkdfz2sm.log",
    "metrics": {},
    "check": {
        "returncode": "N/A",
        "regex": "N/A",
        "runtime": "N/A"
    },
    "tags": "tutorials",
    "starttime": "2021/09/09 15:55:13",
    "endtime": "2021/09/09 15:55:13",
    "runtime": "0.042796",
    "state": "PASS",
    "returncode": "0",
    "output": "1+2=3\nthis is a literal string ':' '\n\nsinglequote'\n\n\"doublequote\"\n\
↪ncurrent user: docs\nnumber of files: 4\n",
    "error": "variables_bash_build.sh: 14: variables_bash_build.sh: source: not\
↪found\n",
    "job": {},
    "build_script": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/7ff48178/variables_
↪bash_build.sh"
  }
]
},
"/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪tutorials/status_regex.yml": {
  "status_regex_pass": [
    {
      "id": "572fadd5",
      "full_id": "572fadd5-b0eb-4df3-aa2f-d878aeb2522d",
      "description": "Pass test based on regular expression",

```

(continues on next page)

(continued from previous page)

```

"schemafile": "script-v1.0.schema.json",
"executor": "generic.local.bash",
"compiler": null,
"hostname": "build-14673784-project-280831-buildtest",
"user": "docs",
"testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5
↳ ",
    "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/
↳ status_regex_pass.sh",
    "stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/
↳ stage",
    "command": "sh status_regex_pass_build.sh",
    "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/status_
↳ regex_pass.out",
    "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/status_
↳ regex_pass.err",
    "buildspec_content": "version: \"1.0\"\\n\\nbuildspecs:\\n  status_regex_pass:\\n  ↵
↳ executor: generic.local.bash\\n  type: script\\n  tags: [system]\\n  description:↵
↳ Pass test based on regular expression\\n  run: echo \"PASS\"\\n  status:\\n  ↵
↳ regex:\\n    stream: stdout\\n    exp: \"^(PASS)$\"\\n\\n  status_regex_fail:\\n  ↵
↳ executor: generic.local.bash\\n  type: script\\n  tags: [system]\\n  description:↵
↳ Pass test based on regular expression\\n  run: echo \"FAIL\"\\n  status:\\n  ↵
↳ regex:\\n    stream: stdout\\n    exp: \"^(123FAIL)$\"\\n\\n",
    "test_content": "#!/bin/bash \\n# Content of run section\\necho \"PASS\"\\n",
    "buildscript_content": "#!/bin/bash\\n\\n##### START VARIABLE↵
↳ DECLARATION #####\\nexport BUILDTEST_TEST_NAME=status_regex_pass\\
↳ nexport BUILDTEST_TEST_ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5\\
↳ nexport BUILDTEST_BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/
↳ buildtest/checkouts/v0.11.0/tutorials\\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ status_regex/status_regex_pass/572fadd5/stage\\nexport BUILDTEST_TEST_ID=572fadd5-b0eb-
↳ 4df3-aa2f-d878aeb2522d\\n##### END VARIABLE DECLARATION #####
↳ ##\\n\\n\\n# source executor startup script\\nsource /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/executor/generic.local.bash/before_script.
↳ sh\\n# Run generated script\\n/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/
↳ stage/status_regex_pass.sh\\n# Get return code\\nreturncode=$?\\n# Exit with return code\\
↳ nexit $returncode",
    "logpath": "/tmp/buildtest_veh1ta6l.log",
    "metrics": {},
    "check": {
        "returncode": false,
        "regex": true,
        "runtime": false
    },
    "tags": "system",

```

(continues on next page)

(continued from previous page)

```

    "starttime": "2021/09/09 15:54:51",
    "endtime": "2021/09/09 15:54:51",
    "runtime": "0.025703",
    "state": "PASS",
    "returncode": "0",
    "output": "PASS\n",
    "error": "status_regex_pass_build.sh: 14: status_regex_pass_build.sh: source:
↳not found\n",
    "job": {},
    "build_script": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/
↳status_regex_pass_build.sh"
  }
],
"status_regex_fail": [
  {
    "id": "099fdf40",
    "full_id": "099fdf40-fac6-4581-ab96-780328380d58",
    "description": "Pass test based on regular expression",
    "schemafile": "script-v1.0.schema.json",
    "executor": "generic.local.bash",
    "compiler": null,
    "hostname": "build-14673784-project-280831-buildtest",
    "user": "docs",
    "testroot": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40
↳",
    "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40/
↳status_regex_fail.sh",
    "stagedir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40/
↳stage",
    "command": "sh status_regex_fail_build.sh",
    "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40/status_
↳regex_fail.out",
    "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40/status_
↳regex_fail.err",
    "buildspec_content": "version: \"1.0\"\\nbuildspecs:\\n  status_regex_pass:\\n
↳executor: generic.local.bash\\n  type: script\\n  tags: [system]\\n  description:
↳Pass test based on regular expression\\n  run: echo \"PASS\"\\n  status:\\n
↳regex:\\n    stream: stdout\\n    exp: \"^(PASS)$\"\\n\\n  status_regex_fail:\\n
↳executor: generic.local.bash\\n  type: script\\n  tags: [system]\\n  description:
↳Pass test based on regular expression\\n  run: echo \"FAIL\"\\n  status:\\n
↳regex:\\n    stream: stdout\\n    exp: \"^(123FAIL)$\"\\n\\n",
    "test_content": "#!/bin/bash \\n# Content of run section\\nnecho \"FAIL\\n\"",
    "buildscript_content": "#!/bin/bash \\n\\n##### START VARIABLE
↳DECLARATION #####\\nexport BUILDTEST_TEST_NAME=status_regex_fail\\
↳nexport BUILDTEST_TEST_ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40\\
↳nexport BUILDTEST_BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/
↳buildtest/checkouts/v0.11.0/tutorials\\nexport BUILDTEST_STAGE_DIR=/home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳status_regex/status_regex_fail/099fdf40/stage\\nexport BUILDTEST_TEST_ID=099fdf40-fac6
↳4581-ab96-780328380d58\\n##### END VARIABLE DECLARATION #####
↳##\\n\\n# source executor startup script\\nsource /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/executor/generic.local.bash/before_script.

```

(continued from previous page)

```

    "logpath": "/tmp/buildtest_veh1ta6l.log",
    "metrics": {},
    "check": {
        "returncode": false,
        "regex": false,
        "runtime": false
    },
    "tags": "system",
    "starttime": "2021/09/09 15:54:51",
    "endtime": "2021/09/09 15:54:51",
    "runtime": "0.025702",
    "state": "FAIL",
    "returncode": "0",
    "output": "FAIL\n",
    "error": "status_regex_fail_build.sh: 14: status_regex_fail_build.sh: source:↵
↵not found\n",
    "job": {},
    "build_script": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↵checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40/
↵status_regex_fail_build.sh"
    }
  ]
}
}

```

Note: If you pass a valid filepath but file is not in cache you will get an error as follows

```

$ buildtest it buildspect $BUILDTEST_ROOT/README.rst
Unable to find any buildspects in cache, please specify one of the following buildspects:
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵vars.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵pass_returncode.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵status_regex.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵runtime_status_test.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵shebang.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵skip_tests.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵metrics_regex.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵metrics_variable.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵executor_regex_script.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵script/multiple_executors.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↵script/executor_scheduler.yml

```

(continues on next page)

(continued from previous page)

```

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪script/status_by_executors.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↪tests/configuration/ulimits.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↪tests/configuration/kernel_state.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↪tests/configuration/disk_usage.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↪tests/configuration/systemd-default-target.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↪tests/configuration/ssh_localhost.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪tags_example.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-hello.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪python-shell.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪run_only_platform.yml
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪hello_world.yml

```

Inspecting Test by ID via `buildtest inspect id`

The `buildtest inspect id` works similar to `buildtest inspect name` except that it operates on test id. This can be useful if you want to extract a particular test record and not see all test records at once.

You only need to specify a few characters and `buildtest` will resolve full test id if there is a match. The `buildtest inspect id` can operate on single or multiple ids if you want to specify multiple ids in single command you can do `buildtest inspect id <identifier1> <identifier2>`.

```

$ buildtest it id a76
{
  "a761ce8f-0442-4ff4-845b-8d7cbd6b563b": {
    "id": "a761ce8f",
    "full_id": "a761ce8f-0442-4ff4-845b-8d7cbd6b563b",
    "description": "hello world example",
    "schemafile": "script-v1.0.schema.json",
    "executor": "generic.local.bash",
    "compiler": null,
    "hostname": "DOE-7086392.local",
    "user": "siddiq90",
    "testroot": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪local.bash/hello_world/hello_world/a761ce8f",
    "testpath": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪local.bash/hello_world/hello_world/a761ce8f/hello_world.sh",
    "stagedir": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪local.bash/hello_world/hello_world/a761ce8f/stage",
    "command": "sh hello_world_build.sh",

```

(continues on next page)

(continued from previous page)

```

"outfile": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/hello_world/hello_world/a761ce8f/hello_world.out",
"errfile": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/hello_world/hello_world/a761ce8f/hello_world.err",
"buildspec_content": "version: \"1.0\"\\nbuidspecs:\\n  hello_world:\\n    executor:
↳ generic.local.bash\\n    type: script\\n    tags: tutorials\\n    description: \"hello_
↳ world example\"\\n    run: echo \"hello world!\"\\n\\nmaintainers:\\n- \"@shahzebsiddiqui\"\\n
↳ ",
"test_content": "#!/bin/bash \\n# Content of run section\\ncho \"hello world!\"\\n",
"buildscript_content": "#!/bin/bash\\n\\n\\n##### START VARIABLE DECLARATION ###
↳ #####\\nexport BUILDTEST_TEST_NAME=hello_world\\nexport BUILDTEST_TEST_
↳ ROOT=/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.bash/
↳ hello_world/hello_world/a761ce8f\\nexport BUILDTEST_BUILDSPEC_DIR=/Users/siddiq90/
↳ Documents/GitHubDesktop/buildtest/tutorials\\nexport BUILDTEST_STAGE_DIR=/Users/
↳ siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.bash/hello_world/
↳ hello_world/a761ce8f/stage\\nexport BUILDTEST_TEST_ID=a761ce8f-0442-4ff4-845b-
↳ 8d7cbd6b563b\\n##### END VARIABLE DECLARATION #####\\n\\n\\n#
↳ source executor startup script\\nsource /Users/siddiq90/Documents/GitHubDesktop/
↳ buildtest/var/executor/generic.local.bash/before_script.sh\\n# Run generated script\\n/
↳ Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.bash/hello_
↳ world/hello_world/a761ce8f/stage/hello_world.sh\\n# Get return code\\nreturncode=$?\\n#
↳ Exit with return code\\nexit $returncode",
"logpath": "/var/folders/lm/_jjv09h17k37mkktwnmbkmj0002t_q/T/buildtest_fcwkh482.log",
"metrics": {},
"tags": "tutorials",
"starttime": "2021/08/19 11:12:32",
"endtime": "2021/08/19 11:12:32",
"runtime": "0.103966",
"state": "PASS",
"returncode": "0",
"output": "hello world!\\n",
"error": "",
"job": {},
"build_script": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/hello_world/hello_world/a761ce8f/hello_world_build.sh"
}
}

```

If you specify an invalid test id using `buildtest inspect id` you will get an error message as follows.

```
$ buildtest inspect id lad
```

```

Unable to find any test records based on id: ['lad'], please run 'buildtest inspect list
↳ ' to see list of ids.

```

You will see similar message if you specify an invalid test name using `buildtest inspect name` command.

Query Test Records via `buildtest inspect query`

The `buildtest inspect query` command can allow you to retrieve query certain fields from each test records that can be useful when you are inspecting a test. Currently, we can fetch content of output file, error file, testpath, and build script. Shown below are the list of available options for `buildtest inspect query`.

```
$ buildtest inspect query --help
usage: buildtest [options] [COMMANDS] inspect query [-h] [-b] [-d {first,last,all}] [-e]
↳ [-o] [-t] [name [name ...]]

positional arguments:
  name                  Name of test

optional arguments:
  -h, --help            show this help message and exit
  -b, --buildscript      Print build script
  -d {first,last,all}, --display {first,last,all}
↳ Determine how records are fetched, by default it will report the
↳ last record of the test.
  -e, --error            Print error file
  -o, --output           Print output file
  -t, --testpath         Print content of testpath
```

The `buildtest inspect query` command expects positional arguments that are name of tests which you can get by running `buildtest inspect list`.

For instance, let's query the test `circle_area` by running the following:

```
$ buildtest inspect query circle_area
----- circle_area (ID: 8e5d3510-bbe7-4140-86d7-aa168c6babec) ----
↳ -----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
returncode: 0
runtime: 0.111269
starttime: 2021/09/09 15:55:22
endtime: 2021/09/09 15:55:22
```

buildtest will display metadata for each test. By default, buildtest will report the latest record for each test that is specified as a positional argument. If you want to see all runs for a particular test you can use `-d all` or `--display all` which will report all records. By default, it will use `-d last` which reports the last record. You can retrieve the first record by running `-d first` which is the oldest record.

Now as you run test, you want to inspect the output file, this can be done by passing `-o` or `--output`. Let's take what we learned and see the following. In this command, we retrieve all records for `circle_area` and print content of output file

```
$ buildtest inspect query -d all -o circle_area
----- circle_area (ID: 6ed8bf63-8008-42c7-9416-7e472c15c9e4) ----
↳ -----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
```

(continues on next page)

(continued from previous page)

```

returncode: 0
runtime: 0.116362
starttime: 2021/09/09 15:55:18
endtime: 2021/09/09 15:55:18
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/6ed8bf63/circle_area.out *****
Circle Radius 2
Area of circle 12.566370614359172

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/6ed8bf63/circle_area.out *****

----- circle_area (ID: 29fd70b3-2bd5-45a0-96af-0b62e394c87e) ----
↳-----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
returncode: 0
runtime: 0.115076
starttime: 2021/09/09 15:55:19
endtime: 2021/09/09 15:55:19
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/29fd70b3/circle_area.out *****
Circle Radius 2
Area of circle 12.566370614359172

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/29fd70b3/circle_area.out *****

----- circle_area (ID: f5c50ca7-2ac3-43b0-a59f-1d88eb00f3de) ----
↳-----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
returncode: 0
runtime: 0.10842
starttime: 2021/09/09 15:55:22
endtime: 2021/09/09 15:55:22
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/f5c50ca7/circle_area.out *****
Circle Radius 2
Area of circle 12.566370614359172

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/f5c50ca7/circle_area.out *****

```

(continues on next page)

(continued from previous page)

```

----- circle_area (ID: 629ee0a6-94c3-4e62-b05c-964f92c15f0a) -----
↳ -----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
returncode: 0
runtime: 0.040307
starttime: 2021/09/09 15:55:22
endtime: 2021/09/09 15:55:22
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳ circle_area/629ee0a6/circle_area.out *****
Circle Radius 2
Area of circle 12.566370614359172

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳ area/629ee0a6/circle_area.out *****

----- circle_area (ID: 8e5d3510-bbe7-4140-86d7-aa168c6babec) -----
↳ -----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
returncode: 0
runtime: 0.111269
starttime: 2021/09/09 15:55:22
endtime: 2021/09/09 15:55:22
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳ circle_area/8e5d3510/circle_area.out *****
Circle Radius 2
Area of circle 12.566370614359172

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳ area/8e5d3510/circle_area.out *****

```

If you want to see content of error file use the `-e` or `--error` flag. It would be useful to inspect content of build script and generated test, which can be retrieved using `--testpath` and `--buildscript`. Let's see query the first record of `circle_area` and report all of the content fields

```

$ buildtest inspect query -d first -o -e -t -b circle_area
----- circle_area (ID: 6ed8bf63-8008-42c7-9416-7e472c15c9e4) -----
↳ -----
executor: generic.local.python
description: Calculate circle of area given a radius
state: PASS
returncode: 0
runtime: 0.116362
starttime: 2021/09/09 15:55:18
endtime: 2021/09/09 15:55:18

```

(continues on next page)

(continued from previous page)

```

***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/6ed8bf63/circle_area.out *****
Circle Radius  2
Area of circle  12.566370614359172

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/6ed8bf63/circle_area.out *****

***** Start of Error File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/6ed8bf63/circle_area.err *****
circle_area_build.sh: 14: circle_area_build.sh: source: not found

***** End of Error File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/6ed8bf63/circle_area.err *****

***** Start of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/6ed8bf63/circle_area.sh *****
#!/bin/bash
python /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳tests/generic.local.python/python-shell/circle_area/6ed8bf63/stage/circle_area.py
***** End of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/6ed8bf63/circle_area.sh *****

***** Start of Build Script: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/6ed8bf63/circle_area_build.sh *****
#!/bin/bash

##### START VARIABLE DECLARATION #####
export BUILDTEST_TEST_NAME=circle_area
export BUILDTEST_TEST_ROOT=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_area/6ed8bf63
export BUILDTEST_BUILDSPEC_DIR=/home/docs/checkouts/readthedocs.org/user_builds/
↳buildtest/checkouts/v0.11.0/tutorials
export BUILDTEST_STAGE_DIR=/home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_area/6ed8bf63/
↳stage
export BUILDTEST_TEST_ID=6ed8bf63-8008-42c7-9416-7e472c15c9e4
##### END VARIABLE DECLARATION #####

# source executor startup script
source /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳executor/generic.local.python/before_script.sh
# Run generated script

```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.python/python-shell/circle_area/6ed8bf63/stage/circle_area.sh
# Get return code
returncode=$?
# Exit with return code
exit $returncode
***** End of Build Script: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/
↳circle_area/6ed8bf63/circle_area_build.sh *****
```

We can query multiple tests using `buildtest inspect query` since each test is a positional argument. Any options specified to `buildtest inspect query` will be applied to all test. For instance, let's fetch the output the of test names `root_disk_usage` and `python_hello`

```
$ buildtest inspect query -o root_disk_usage python_hello
----- root_disk_usage (ID: 1cd41359-a069-43ea-8807-
↳2b6f256c34e3) -----
executor: generic.local.bash
description: Check root disk usage and report if it exceeds threshold
state: PASS
returncode: 0
runtime: 0.037134
starttime: 2021/09/09 15:55:14
endtime: 2021/09/09 15:55:14
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_
↳disk_usage/1cd41359/root_disk_usage.out *****
[OK] Root disk is below threshold of 90%

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/disk_usage/root_disk_
↳usage/1cd41359/root_disk_usage.out *****

----- python_hello (ID: 3c422a2b-81b4-4d4f-9240-1e74655d9c8e) --
↳-----
executor: generic.local.bash
description: Hello World python
state: PASS
returncode: 0
runtime: 0.092008
starttime: 2021/09/09 15:55:18
endtime: 2021/09/09 15:55:18
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/
↳python_hello/3c422a2b/python_hello.out *****
Hello World

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
↳hello/3c422a2b/python_hello.out *****
```

Using Alternate Report File

The `buildtest report` and `buildtest inspect` command will read from the report file tracked by buildtest which is stored in `$BUILDTEST_ROOT/var/report.json`. This single file can become an issue if you are running jobs through CI where you can potentially overwrite same file or if you want separate report files for each set of builds. Luckily we have an option to handle this using the `buildtest build -r /path/to/report` option which can be used to specify an alternate location to report file.

buildtest will write the report file in the desired location, then you can specify the path to report file via `buildtest report -r /path/to/report` and `buildtest inspect -r /path/to/report` to load the report file when reporting tests.

The report file must be valid JSON file that buildtest understands in order to use *buildtest report* and *buildtest inspect* command. Shown below are some examples using the alternate report file using `buildtest report` and `buildtest inspect` command.

```
$ buildtest report -r python.json --format name,id
Reading report file: /Users/siddiq90/Documents/GitHubDesktop/buildtest/docs/python.json
```

```
+-----+-----+
| name      | id        |
+=====+=====+
| circle_area | 6be6c404 |
+-----+-----+
| python_hello | f21ba744 |
+-----+-----+
```

```
$ buildtest inspect -r test.json name variables_bash
Reading Report File: /Users/siddiq90/Documents/GitHubDesktop/buildtest/test.json
```

```
{
  "variables_bash": [
    {
      "id": "cd0511ce",
      "full_id": "cd0511ce-377e-4ed2-95f4-f244e5518732",
      "schemafilename": "script-v1.0.schema.json",
      "executor": "generic.local.bash",
      "compiler": null,
      "hostname": "DOE-7086392.local",
      "user": "siddiq90",
      "testroot": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/
↪ variables_bash/1",
      "testpath": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/
↪ variables_bash/1/stage/generate.sh",
      "stagedir": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/
↪ variables_bash/1/stage",
      "rundir": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/1/run",
      "command": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/1/stage/generate.sh",
      "outfile": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/1/run/variables_bash.out",
      "errfile": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/1/run/variables_bash.err",
```

(continues on next page)

(continued from previous page)

```

    "buildspec_content": "version: \"1.0\"\nbuildspecs:\n  variables_bash:\n    type: \u2192script\n    executor: generic.local.bash\n    description: Declare shell variables in.\n    bash:\n      tags: [tutorials]\n      vars:\n        X: 1\n        Y: 2\n        literalstring: |\n        \"this is a literal string ':' '\n        singlequote: '\"singlequote'\n        doublequote: '\"\"doublequote\"\"'\n        current_user: \"$(whoami)\"\n        files_\n    homedir: \"`find $HOME -type f -maxdepth 1`\"\n        run: |\n        echo \"$X+$Y=\" \"${(\n    $X+$Y))}\n        echo $literalstring\n        echo $singlequote\n        echo $doublequote\n\n        echo $current_user\n        echo $files_homedir",
    "test_content": "#!/bin/bash\nsource /Users/siddiq90/.buildtest/executor/generic.\nlocal.bash/before_script.sh\nX=1\nY=2\nliteralstring=\"this is a literal string ':' '\n\nsinglequote='singlequote'\ndoublequote=\"doublequote\"\ncurrent_user=$(whoami)\nfiles_homedir=`find $HOME -type f -maxdepth 1`\necho \"$X+$Y=\" \"${($X+$Y)}\necho\n$literalstring\necho $singlequote\necho $doublequote\n\necho $current_user\necho\n$files_homedir\nsource /Users/siddiq90/.buildtest/executor/generic.local.bash/after_\nscript.sh",
    "tags": "tutorials",
    "starttime": "2021/04/16 14:29:25",
    "endtime": "2021/04/16 14:29:25",
    "runtime": 0.213196,
    "state": "PASS",
    "returncode": 0,
    "output": "1+2= 3\nthis is a literal string ':'\nsinglequote\ndoublequote\n\nsiddiq90\n/Users/siddiq90/buildtest_e7yxgttm.log /Users/siddiq90/.anyconnect /Users/\nsiddiq90/buildtest_utwigh8w.log /Users/siddiq90/.DS_Store /Users/siddiq90/.serverauth.\n555 /Users/siddiq90/.CFUserTextEncoding /Users/siddiq90/.wget-hsts /Users/siddiq90/.\nbashrc /Users/siddiq90/.zshrc /Users/siddiq90/.coverage /Users/siddiq90/.serverauth.\n87055 /Users/siddiq90/buildtest_r7bck5zh.log /Users/siddiq90/.zsh_history /Users/\nsiddiq90/.lessht /Users/siddiq90/calltracker.py /Users/siddiq90/.git-completion.bash /\nUsers/siddiq90/buildtest_wvjaaztp.log /Users/siddiq90/buildtest.log /Users/siddiq90/\ndarhan.log /Users/siddiq90/ascent.yml /Users/siddiq90/.cshrc /Users/siddiq90/buildtest_\nnyq22whj.log /Users/siddiq90/github-tokens /Users/siddiq90/buildtest_ozb8b52z.log /\nUsers/siddiq90/.zcompdump /Users/siddiq90/buildtest_nab_ckph.log /Users/siddiq90/.\nserverauth.543 /Users/siddiq90/.s.PGSQL.15007.lock /Users/siddiq90/.bash_profile /\nUsers/siddiq90/.Xauthority /Users/siddiq90/.python_history /Users/siddiq90/.gitconfig /\nUsers/siddiq90/output.txt /Users/siddiq90/.bash_history /Users/siddiq90/.viminfo\n",
    "error": "",
    "job": null
  },
  {
    "id": "e0901505",
    "full_id": "e0901505-a66b-4c91-9b29-d027cb6fabb6",
    "schemafilename": "script-v1.0.schema.json",
    "executor": "generic.local.bash",
    "compiler": null,
    "hostname": "DOE-7086392.local",
    "user": "siddiq90",
    "testroot": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/\nvariables_bash/2",
    "testpath": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/\nvariables_bash/2/stage/generate.sh",
    "stagedir": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/\nvariables_bash/2/stage",

```

(continues on next page)

(continued from previous page)

```

    "rundir": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/2/run",
    "command": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/2/stage/generate.sh",
    "outfile": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/2/run/variables_bash.out",
    "errfile": "/Users/siddiq90/.buildtest/var/tests/generic.local.bash/vars/variables_
↪ bash/2/run/variables_bash.err",
    "buildspec_content": "version: \"1.0\"\\nbuildspecs:\\n  variables_bash:\\n    type:␣
↪ script\\n    executor: generic.local.bash\\n    description: Declare shell variables in␣
↪ bash\\n    tags: [tutorials]\\n    vars:\\n      X: 1\\n      Y: 2\\n      literalstring: |\\
↪ \\n      \\\"this is a literal string ':' '\\n      singlequote: '\\\"singlequote'\\\"\\n      ␣
↪ doublequote: '\\\"\\\"doublequote\\\"\\\"\\\"\\n      current_user: '\\$(whoami)'\\\"\\n      files_
↪ homedir: '\\`find $HOME -type f -maxdepth 1`\\\"\\n      run: |\\n      echo '\\$X+$Y=\\\" $((
↪ $X+$Y))\\n      echo $literalstring\\n      echo $singlequote\\n      echo $doublequote\\n\\
↪ \\n      echo $current_user\\n      echo $files_homedir",
    "test_content": "#!/bin/bash \\nsource /Users/siddiq90/.buildtest/executor/generic.
↪ local.bash/before_script.sh\\nX=1\\nY=2\\nliteralstring=\\\"this is a literal string ':' '\\
↪ \\n\\nsinglequote='singlequote'\\ndoublequote=\\\"doublequote\\\"\\ncurrent_user=$(whoami)\\
↪ \\nfiles_homedir=`find $HOME -type f -maxdepth 1`\\necho '\\$X+$Y=\\\" $((($X+$Y))\\necho
↪ $literalstring\\necho $singlequote\\necho $doublequote\\n\\necho $current_user\\necho
↪ $files_homedir\\nsource /Users/siddiq90/.buildtest/executor/generic.local.bash/after_
↪ script.sh",
    "tags": "tutorials",
    "starttime": "2021/04/16 14:29:58",
    "endtime": "2021/04/16 14:29:58",
    "runtime": 0.075224,
    "state": "PASS",
    "returncode": 0,
    "output": "1+2= 3\\nthis is a literal string ':'\\nsinglequote\\ndoublequote\\
↪ nsiddiq90\\n/Users/siddiq90/buildtest_e7yxgttm.log /Users/siddiq90/.anyconnect /Users/
↪ siddiq90/buildtest_utwigb8w.log /Users/siddiq90/.DS_Store /Users/siddiq90/.serverauth.
↪ 555 /Users/siddiq90/.CFUserTextEncoding /Users/siddiq90/.wget-hsts /Users/siddiq90/.
↪ bashrc /Users/siddiq90/.zshrc /Users/siddiq90/.coverage /Users/siddiq90/.serverauth.
↪ 87055 /Users/siddiq90/buildtest_r7bck5zh.log /Users/siddiq90/.zsh_history /Users/
↪ siddiq90/.lessht /Users/siddiq90/calltracker.py /Users/siddiq90/.git-completion.bash /
↪ Users/siddiq90/buildtest_wvjaaztp.log /Users/siddiq90/buildtest.log /Users/siddiq90/
↪ darhan.log /Users/siddiq90/ascent.yml /Users/siddiq90/.cshrc /Users/siddiq90/buildtest_
↪ nyq22whj.log /Users/siddiq90/github-tokens /Users/siddiq90/buildtest_ozb8b52z.log /
↪ Users/siddiq90/.zcompdump /Users/siddiq90/buildtest_nab_ckph.log /Users/siddiq90/.
↪ serverauth.543 /Users/siddiq90/.s.PGSQL.15007.lock /Users/siddiq90/.bash_profile /
↪ Users/siddiq90/.Xauthority /Users/siddiq90/.python_history /Users/siddiq90/.gitconfig /
↪ Users/siddiq90/output.txt /Users/siddiq90/.bash_history /Users/siddiq90/.viminfo\\n",
    "error": "",
    "job": null
  }
]
}

```


3.4 Buildspec Tutorial

3.4.1 Buildspec Overview

What is a buildspec?

In buildtest, we refer to **buildspec** as a YAML file that defines your test that buildtest will parse using the provided schemas and build a shell script from the buildspec file. Every buildspec is validated with a global schema which you can find more if you click [here](#).

Example

Let's start off with a simple example that declares two variables **X** and **Y** and prints the sum of X+Y.

```
version: "1.0"
buildspecs:
  add_numbers:
    type: script
    executor: generic.local.bash
    description: Add X+Y
    tags: [tutorials]
    vars:
      X: 1
      Y: 2
    run: echo "$X+$Y=" $((X+Y))
```

buildtest will validate the entire file with `global.schema.json`, the schema requires **version** and **buildspec** in order to validate file. The **buildspec** is where you define each test. The name of the test is **add_numbers**. The test requires a **type** field which is the sub-schema used to validate the test section. In this example `type: script` informs buildtest to use the *Script Schema* when validating test section.

Each subschema has a list of field attributes that are supported, for example the fields: **type**, **executor**, **vars** and **run** are all valid fields supported by the *script* schema. The **version** field informs which version of subschema to use. Currently all sub-schemas are at version 1.0 where buildtest will validate with a schema `script-v1.0.schema.json`. In future, we can support multiple versions of subschema for backwards compatibility.

Let's look at a more interesting example, shown below is a multi line run example using the *script* schema with test name called **systemd_default_target**, shown below is the content of test:

```
version: "1.0"
buildspecs:
  systemd_default_target:
    executor: generic.local.bash
    type: script
    tags: [system]
    description: check if default target is multi-user.target
    run: |
      if [ "multi-user.target" == `systemctl get-default` ]; then
        echo "multi-user is the default target";
        exit 0
      fi
      echo "multi-user is not the default target";
      exit 1
```

The test name `systemd_default_target` defined in `buildspec` section is validated with the following pattern `^[A-Za-z_][A-Za-z0-9_]*$`. This test will use the executor `generic.local.bash` which means it will use the Local Executor with an executor name `bash` defined in the buildtest settings. The default buildtest settings will provide a bash executor as follows:

```
system:
  generic:
    hostnames: ["localhost"]
    executors:
      local:
        bash:
          description: submit jobs on local machine using bash shell
          shell: bash
```

The `shell: bash` indicates this executor will use `bash` to run the test scripts. To reference this executor use the format `<system>.<type>.<name>` in this case `generic.local.bash` refers to bash executor.

The `description` field is an optional key that can be used to provide a brief summary of the test. In this example we can a full multi-line run section, this is achieved in YAML using `run:` followed by content of run section tab indented 2 spaces.

Script Schema

The script schema is used for writing simple scripts (bash, sh, python) in Buildspec. To use this schema you must set `type: script`. The `run` field is responsible for writing the content of test.

Shown below is schema header for `script-v1.0.schema.json`.

```
{
  "$id": "script-v1.0.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "script schema version 1.0",
  "description": "The script schema is of `type: script` in sub-schema which is used_
↪for running shell scripts",
  "type": "object",
  "required": ["type", "run", "executor"],
  "additionalProperties": false,
```

The `"type": "object"` means sub-schema is a JSON `object` where we define a list of key/value pair. The `"required"` field specifies a list of fields that must be specified in order to validate the Buildspec. In this example, `type`, `run`, and `executor` are required fields. The `additionalProperties: false` informs schema to reject any extra properties not defined in the schema.

The `executor` key is required for all sub-schemas which instructs buildtest which executor to use when running the test. The executors are defined in *Configuring buildtest*. In our *first example* we define variables using the `vars` property which is a Key/Value pair for variable assignment. The `run` section is required for script schema which defines the content of the test script.

For more details on script schema see schema docs at <https://buildtesters.github.io/buildtest/>

Declaring Environment Variables

You can define environment variables using the `env` property, this is compatible with shells: `bash`, `sh`, `zsh`, `cs`h and `tcsh`. It does not work with `shell: python`. In example below we declare three tests using environment variable with default shell (`bash`), `cs`h, and `tcsh`

```
version: "1.0"
buildspecs:
  bash_env_variables:
    executor: generic.local.bash
    description: Declare environment variables in default shell (bash)
    type: script
    env:
      FIRST_NAME: avocado
      LAST_NAME: dinosaur
    tags: [tutorials]
    run: |
      hostname
      whoami
      echo $USER
      printf "${FIRST_NAME} ${LAST_NAME}\n"

  csh_env_declaration:
    executor: generic.local.csh
    type: script
    description: "csh shell example to declare environment variables"
    shell: /bin/csh
    tags: [tutorials]
    env:
      SHELL_NAME: "csh"
    run: echo "This is running $SHELL_NAME"

  tcsh_env_declaration:
    executor: generic.local.csh
    type: script
    description: "tcsh shell example to declare environment variables"
    shell: /bin/tcsh
    tags: [tutorials]
    env:
      path: "/usr/local/bin:$PATH"
    run: echo $path
```

This test can be run by issuing the following command: `buildtest build -b tutorials/environment.yml`. If we inspect one of the test script we will see that buildtest generates a build script that invokes the test using the shell wrapper `/bin/csh` for the `csh` test and gets the returncode.

```
#!/bin/bash

##### START VARIABLE DECLARATION #####
export BUILDTEST_TEST_NAME=csh_env_declaration
export BUILDTEST_TEST_ROOT=/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↳generic.local.csh/environment/csh_env_declaration/0
export BUILDTEST_BUILDSPEC_DIR=/Users/siddiq90/Documents/GitHubDesktop/buildtest/
↳tutorials
```

(continues on next page)

(continued from previous page)

```

export BUILDTEST_STAGE_DIR=/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↳generic.local.csh/environment/csh_env_declaration/0/stage
export BUILDTEST_TEST_ID=501ec5d3-e614-4ae8-9c1e-4849ce340c76
##### END VARIABLE DECLARATION #####

# source executor startup script
source /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/executor/generic.local.csh/
↳before_script.sh
# Run generated script
/bin/csh /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.csh/
↳environment/csh_env_declaration/0/stage/csh_env_declaration.csh
# Get return code
returncode=$?
# Exit with return code
exit $returncode

```

This generated test looks something like this

```

#!/bin/csh
# Declare environment variables
setenv SHELL_NAME csh

# Content of run section
echo "This is running $SHELL_NAME"

```

Environment variables are defined using `export` in bash, sh, zsh while csh and tcsh use `setenv`.

Declaring Variables

Variables can be defined using `vars` property, this is compatible with all shells except for `python`. The variables are defined slightly different in csh, tcsh as pose to bash, sh, and zsh. In example below we define tests with bash and csh.

In YAML strings can be specified with or without quotes however in bash, variables need to be enclosed in quotes " if you are defining a multi word string (`name="First Last"`).

If you need define a literal string it is recommended to use the literal block `|` that is a special character in YAML. If you want to specify " or ' in string you can use the escape character `\` followed by any of the special character. In example below we define several variables such as `X`, `Y` that contain numbers, variable `literalstring` is a literal string processed by YAML. The variable `singlequote` and `doublequote` defines a variable with the special character ' and ". The variables `current_user` and `num_files` store result of a shell command. This can be done using `var=$(<command>)` or `var=` <command> `` where `<command>` is a Linux command.

```

version: "1.0"
buildspecs:
  variables_bash:
    type: script
    executor: generic.local.bash
    description: Declare shell variables in bash
    tags: [tutorials]
    vars:
      X: 1

```

(continues on next page)

(continued from previous page)

```

Y: 2
literalstring: |
    "this is a literal string ':' "
singlequote: \'singlequote\'
doublequote: \"doublequote\"
current_user: \"$(whoami)\"
num_files: "`find $HOME -type f -maxdepth 1 | wc -l`"

run: |
    echo "$X+$Y=$((X+$Y))"
    echo $literalstring
    echo $singlequote
    echo $doublequote
    echo "current user:" $current_user
    echo "number of files:" $num_files

```

Next we build this test by running `buildtest build -b $BUILDTEST_ROOT/tutorials/vars.yml`.

```

$ buildtest build -b $BUILDTEST_ROOT/tutorials/vars.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:54:50
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/tutorials/vars.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ vars.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0

```

(continues on next page)

(continued from previous page)

Detected Buildsspecs after exclusion: 1

```

+-----+
| Stage: Parsing Buildsspecs |
+-----+

```

Valid Buildsspecs: 1

Invalid Buildsspecs: 0

```

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪vars.yml: VALID

```

Total builder objects created: 1

builders: [variables_bash/98b07db8]

name	id	description	buildspecs
------	----	-------------	------------

```

↪-----↪
variables_bash 98b07db8 Declare shell variables in bash /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml

```

```

+-----+
| Stage: Building Test |
+-----+

```

name	id	type	executor	tags	testpath
------	----	------	----------	------	----------

```

↪-----↪
variables_bash | 98b07db8 | script | generic.local.bash | ['tutorials'] | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↪local.bash/vars/variables_bash/98b07db8/variables_bash_build.sh

```

```

+-----+
| Stage: Running Test |
+-----+

```

variables_bash/98b07db8: completed with returncode: 0

```

variables_bash/98b07db8: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↪builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↪98b07db8/variables_bash.out

```

```

variables_bash/98b07db8: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↪builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↪98b07db8/variables_bash.err

```

Launching test: variables_bash

Test ID: 98b07db8-9b3e-4ff0-b526-1ebcebbc3a1e

Executor Name: generic.local.bash

```

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash_build.sh

```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Test Summary |
+-----+

name          | id          | executor          | status | returncode_match | regex_
↪match      | runtime_match | returncode | runtime
+-----+-----+-----+-----+-----+-----+
↪-----+-----+-----+-----+-----+-----+
variables_bash | 98b07db8 | generic.local.bash | PASS   | N/A              | N/A ↪
↪          | N/A      |          | 0 | 0.012987

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_8xn_louq.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

```

Let's check the generated script from the previous build, you can run `buildtest inspect query -o variables_bash` where `-o` refers to output file for testname `variables_bash`. Take note of the output file we

```

$ buildtest inspect query -o variables_bash
----- variables_bash (ID: 98b07db8-9b3e-4ff0-b526-1ebcebbc3a1e) ↪
↪-----
executor: generic.local.bash
description: Declare shell variables in bash
state: PASS
returncode: 0
runtime: 0.012987
starttime: 2021/09/09 15:54:50
endtime: 2021/09/09 15:54:50
***** Start of Output File: /home/docs/checkouts/readthedocs.org/
↪user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_
↪bash/98b07db8/variables_bash.out *****
1+2=3
this is a literal string ':'
'singlequote'
"doublequote"
current user: docs
number of files: 4

***** End of Output File: /home/docs/checkouts/readthedocs.org/user_
↪builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/vars/variables_bash/
↪98b07db8/variables_bash.out *****

```

Test Status

buildtest will record state of each test which can be PASS or FAIL. By default a 0 exit code is PASS and everything else is a FAIL. The status property can be used to determine how test will report its state. Currently, we can match state based on *returncode*, *runtime*, or *regular expression*.

Return Code Matching

buildtest can report PASS/FAIL based on returncode, by default a 0 exit code is PASS and everything else is FAIL. The returncode can be a list of exit codes to match. In this example we have four tests called `exit1_fail`, `exit1_pass`, `returncode_list_mismatch` and `returncode_int_match`. We expect `exit1_fail` and `returncode_mismatch` to FAIL while `exit1_pass` and `returncode_int_match` will PASS.

```
version: "1.0"
```

```
buildspecs:
```

```
  exit1_fail:
```

```
    executor: generic.local.sh
```

```
    type: script
```

```
    description: exit 1 by default is FAIL
```

```
    tags: [tutorials, fail]
```

```
    run: exit 1
```

```
  exit1_pass:
```

```
    executor: generic.local.sh
```

```
    type: script
```

```
    description: report exit 1 as PASS
```

```
    run: exit 1
```

```
    tags: [tutorials, pass]
```

```
    status:
```

```
      returncode: [1]
```

```
  returncode_list_mismatch:
```

```
    executor: generic.local.sh
```

```
    type: script
```

```
    description: exit 2 failed since it failed to match returncode 1
```

```
    run: exit 2
```

```
    tags: [tutorials, fail]
```

```
    status:
```

```
      returncode: [1, 3]
```

```
  returncode_int_match:
```

```
    executor: generic.local.sh
```

```
    type: script
```

```
    description: exit 128 matches returncode 128
```

```
    run: exit 128
```

```
    tags: [tutorials, pass]
```

```
    status:
```

```
      returncode: 128
```

Let's build this test and pay close attention to the **status** column in output.


```
$ buildtest build -b tutorials/pass_returncode.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:54:51
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/pass_returncode.yml
```

```
+-----+
| Stage: Discovering Buildsspecs |
+-----+
```

```
+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml |
+-----+
```

```
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1
```

```
+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

```
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml: VALID
```

```
Total builder objects created: 4
builders: [exit1_fail/53c2663d, exit1_pass/97c4d576, returncode_list_mismatch/31d04ee2,
↳ returncode_int_match/278c66df]
```

```
name          id          description
↳ buildspecs
```

(continues on next page)

(continued from previous page)

```

exit1_fail          53c2663d  exit 1 by default is FAIL
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml
exit1_pass          97c4d576  report exit 1 as PASS
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml
returncode_list_mismatch 31d04ee2  exit 2 failed since it failed to match returncode 1
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml
returncode_int_match 278c66df  exit 128 matches returncode 128
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ pass_returncode.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags
↳ | testpath
-----+-----+-----+-----+-----
↳ +-----+-----+-----+-----+-----
↳ -----
↳ -----
exit1_fail          | 53c2663d | script | generic.local.sh | ['tutorials', 'fail']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/exit1_fail/53c2663d/exit1_fail_build.sh
exit1_pass          | 97c4d576 | script | generic.local.sh | ['tutorials', 'pass']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/exit1_pass/97c4d576/exit1_pass_build.sh
returncode_list_mismatch | 31d04ee2 | script | generic.local.sh | ['tutorials', 'fail']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/returncode_list_mismatch/31d04ee2/returncode_
↳ list_mismatch_build.sh
returncode_int_match      | 278c66df | script | generic.local.sh | ['tutorials', 'pass']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/pass_returncode/returncode_int_match/278c66df/returncode_int_
↳ match_build.sh

+-----+
| Stage: Running Test |
+-----+

exit1_pass/97c4d576: completed with returncode: 1
exit1_pass/97c4d576: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/97c4d576/exit1_pass.out
exit1_pass/97c4d576: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳ pass/97c4d576/exit1_pass.err
exit1_pass/97c4d576: Checking returncode - 1 is matched in list [1]
returncode_list_mismatch/31d04ee2: completed with returncode: 2
returncode_list_mismatch/31d04ee2: Writing output file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳ returncode_list_mismatch/31d04ee2/returncode_list_mismatch.out

```

(continued from previous page)

```

returncode_list_mismatch/31d04ee2: Writing error file: /home/docs/checkouts/readthedocs.
↳org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_list_mismatch/31d04ee2/returncode_list_mismatch.err
returncode_list_mismatch/31d04ee2: Checking returncode - 2 is matched in list [1, 3]
exit1_fail/53c2663d: completed with returncode: 1
exit1_fail/53c2663d: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳fail/53c2663d/exit1_fail.out
exit1_fail/53c2663d: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/exit1_
↳fail/53c2663d/exit1_fail.err
returncode_int_match/278c66df: completed with returncode: 128
returncode_int_match/278c66df: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_int_match/278c66df/returncode_int_match.out
returncode_int_match/278c66df: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/pass_returncode/
↳returncode_int_match/278c66df/returncode_int_match.err
returncode_int_match/278c66df: Checking returncode - 128 is matched in list [128]

```

Launching test: exit1_fail

Test ID: 53c2663d-6290-4715-bb97-eb2f8b99ff86

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/exit1_fail/53c2663d/exit1_fail_build.sh

Launching test: exit1_pass

Test ID: 97c4d576-0e99-4f02-abdc-aa8b54802834

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/exit1_pass/97c4d576/exit1_pass_build.sh

Launching test: returncode_list_mismatch

Test ID: 31d04ee2-707e-441a-ad5e-a3cdb862860e

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_list_mismatch/31d04ee2/
↳returncode_list_mismatch_build.sh

Launching test: returncode_int_match

Test ID: 278c66df-fc80-4a90-9968-fc74811c5f7f

Executor Name: generic.local.sh

Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/pass_returncode/returncode_int_match/278c66df/
↳returncode_int_match_build.sh

+-----+
| Stage: Test Summary |
+-----+

name	id	executor	status	returncode_match
↳ regex_match	runtime_match	returncode	runtime	

(continues on next page)

(continued from previous page)

exit1_fail		53c2663d	generic.local.sh	FAIL	N/A	
↪	N/A	N/A		1	0.023217	
exit1_pass		97c4d576	generic.local.sh	PASS	True	
↪	False	False		1	0.020955	
returncode_list_mismatch		31d04ee2	generic.local.sh	FAIL	False	
↪	False	False		2	0.019587	
returncode_int_match		278c66df	generic.local.sh	PASS	True	
↪	False	False		128	0.019961	

Passed Tests: 2/4 Percentage: 50.000%

Failed Tests: 2/4 Percentage: 50.000%

Writing Logfile to: /tmp/buildtest_s4hf3x_d.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

The `returncode` field can be an integer or list of integers but it may not accept duplicate values. If you specify a list of exit codes, buildtest will check actual returncode with list of expected returncodes specified by `returncode` field.

Shown below are examples of invalid returncodes:

```
# empty list is not allowed
returncode: []

# floating point is not accepted in list
returncode: [1, 1.5]

# floating point not accepted
returncode: 1.5

# duplicates are not allowed
returncode: [1, 2, 5, 5]
```

Passing Test based on regular expression

buildtest can configure PASS/FAIL of test based on regular expression on output or error file. This can be useful if you are expecting a certain output from the test as pose to returncode check.

In this example we introduce, the `regex` field which is part of **status** that expects a regular expression via `exp`. The stream property must be **stdout** or **stderr** which indicates buildtest will read output or error file and apply regular expression. If there is a match, buildtest will record the test state as **PASS** otherwise it will be a **FAIL**. In this example, we have two tests that will apply regular expression on output file.

```
version: "1.0"
buildspecs:
  status_regex_pass:
    executor: generic.local.bash
```

(continues on next page)

(continued from previous page)

```

type: script
tags: [system]
description: Pass test based on regular expression
run: echo "PASS"

```

```

status:
  regex:
    stream: stdout
    exp: "^(PASS)$"

```

```

status_regex_fail:
  executor: generic.local.bash
  type: script
  tags: [system]
  description: Pass test based on regular expression
  run: echo "FAIL"
  status:
    regex:
      stream: stdout
      exp: "^(123FAIL)$"

```

Now if we run this test, we will see first test will pass while second one will fail even though the returncode is a 0. Take a close look at the **status** property

```

$ buildtest build -b tutorials/status_regex.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:54:51
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/status_regex.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ status_regex.yml |

```

(continues on next page)

(continued from previous page)

```

+-----+
↪-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪status_regex.yml: VALID

Total builder objects created: 2
builders: [status_regex_pass/572fadd5, status_regex_fail/099fdf40]

name            id            description                                     buildsspecs
-----
↪-----
status_regex_pass 572fadd5 Pass test based on regular expression /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/status_regex.yml
status_regex_fail 099fdf40 Pass test based on regular expression /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/status_regex.yml

+-----+
| Stage: Building Test |
+-----+

name            | id            | type    | executor          | tags          | testpath
-----+-----+-----+-----+-----+-----
↪-----
↪-----
status_regex_pass | 572fadd5 | script | generic.local.bash | ['system'] | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↪local.bash/status_regex/status_regex_pass/572fadd5/status_regex_pass_build.sh
status_regex_fail | 099fdf40 | script | generic.local.bash | ['system'] | /home/docs/
↪checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↪local.bash/status_regex/status_regex_fail/099fdf40/status_regex_fail_build.sh

+-----+
| Stage: Running Test |
+-----+

status_regex_fail/099fdf40: completed with returncode: 0
status_regex_fail/099fdf40: Writing output file: /home/docs/checkouts/readthedocs.org/
↪user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/
↪status_regex_fail/099fdf40/status_regex_fail.out
status_regex_fail/099fdf40: Writing error file: /home/docs/checkouts/readthedocs.org/
↪user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/
↪status_regex_fail/099fdf40/status_regex_fail.err

```

(continued from previous page)

```

status_regex_fail/099fdf40: performing regular expression - '^(123FAIL)$' on file: /home/
↳ docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳ generic.local.bash/status_regex/status_regex_fail/099fdf40/status_regex_fail.out with
↳ regular expression
status_regex_fail/099fdf40: Regular Expression Match - Failed!
status_regex_pass/572fadd5: completed with returncode: 0
status_regex_pass/572fadd5: Writing output file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/
↳ status_regex_pass/572fadd5/status_regex_pass.out
status_regex_pass/572fadd5: Writing error file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/status_regex/
↳ status_regex_pass/572fadd5/status_regex_pass.err
status_regex_pass/572fadd5: performing regular expression - '^(PASS)$' on file: /home/
↳ docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳ generic.local.bash/status_regex/status_regex_pass/572fadd5/status_regex_pass.out with
↳ regular expression
status_regex_pass/572fadd5: Regular Expression Match - Success!

```

```

-----
Launching test: status_regex_pass
Test ID: 572fadd5-b0eb-4df3-aa2f-d878aeb2522d
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/status_regex/status_regex_pass/572fadd5/status_regex_
↳ pass_build.sh

```

```

-----
Launching test: status_regex_fail
Test ID: 099fdf40-fac6-4581-ab96-780328380d58
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/status_regex/status_regex_fail/099fdf40/status_regex_
↳ fail_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_match	
↳ regex_match	↳ runtime_match	↳ returncode	↳ runtime		
status_regex_pass	572fadd5	generic.local.bash	PASS	False	↳
↳ True	False	0	0.025703		
status_regex_fail	099fdf40	generic.local.bash	FAIL	False	↳
↳ False	False	0	0.025702		

```

Passed Tests: 1/2 Percentage: 50.000%
Failed Tests: 1/2 Percentage: 50.000%

```

```

Writing Logfile to: /tmp/buildtest_veh1ta6l.log

```

(continues on next page)

(continued from previous page)

A copy of logfile can be found at \$BUILDTTEST_ROOT/buildtest.log - /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

Passing Test based on runtime

buildtest can determine state of test based on *runtime* property which is part of *status* object. This can be used if you want to control how test *PASS* or *FAIL* based on execution time of test. In example below we have five tests that make use of **runtime** property for passing a test. The runtime property support *min* and *max* property that can mark test pass based on minimum and maximum runtime. A test will pass if it's execution time is greater than *min* time and less than *max* time. If *min* is specified without *max* property the upperbound is not set, likewise *max* without *min* will pass if test is less than **max** time. The lower bound is not set, but test runtime will be greater than 0 sec.

In test **timelimit_min**, we sleep for 2 seconds and it will pass because minimum runtime is 1.0 seconds. Similarly, **timelimit_max** will pass because we sleep for 2 seconds with a max time of 5.0.

```
version: "1.0"
buildspecs:
  timelimit_min_max:
    type: script
    executor: generic.local.sh
    description: "Run a sleep job for 2 seconds and test pass if its within 1.0-3.0sec"
    tags: ["tutorials"]
    run: sleep 2
    status:
      runtime:
        min: 1.0
        max: 3.0

  timelimit_min:
    type: script
    executor: generic.local.sh
    description: "Run a sleep job for 2 seconds and test pass if its exceeds min time of 1.0 sec"
    tags: ["tutorials"]
    run: sleep 2
    status:
      runtime:
        min: 1.0

  timelimit_max:
    type: script
    executor: generic.local.sh
    description: "Run a sleep job for 2 seconds and test pass if it's within max time: 5.0 sec"
    tags: ["tutorials"]
    run: sleep 2
    status:
      runtime:
        max: 5.0
```

(continues on next page)

(continued from previous page)

```

timelimit_min_fail:
  type: script
  executor: generic.local.sh
  description: "This test fails because it runs less than mintime of 10 second"
  tags: ["tutorials"]
  run: sleep 2
  status:
    runtime:
      min: 10.0

timelimit_max_fail:
  type: script
  executor: generic.local.sh
  description: "This test fails because it exceeds maxtime of 1.0 second"
  tags: ["tutorials"]
  run: sleep 3
  status:
    runtime:
      max: 1.0

```

```

$ buildtest build -b tutorials/runtime_status_test.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:54:52
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/runtime_status_test.yml

```

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ runtime_status_test.yml |
+-----+
↳ -----+

```

(continues on next page)

(continued from previous page)

```

Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳runtime_status_test.yml: VALID

Total builder objects created: 5
builders: [timelimit_min_max/bf6aaa60, timelimit_min/810cab9e, timelimit_max/8385a7e5,
↳timelimit_min_fail/5352acc1, timelimit_max_fail/40bee7c5]

name          id          description
↳           buildsspecs
-----
↳
↳
timelimit_min_max bf6aaa60 Run a sleep job for 2 seconds and test pass if its within
↳1.0-3.0sec /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml
timelimit_min      810cab9e Run a sleep job for 2 seconds and test pass if its exceeds
↳min time of 1.0 sec /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml
timelimit_max      8385a7e5 Run a sleep job for 2 seconds and test pass if it's within
↳max time: 5.0 sec /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml
timelimit_min_fail 5352acc1 This test fails because it runs less than mintime of 10
↳second /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml
timelimit_max_fail 40bee7c5 This test fails because it exceeds maxtime of 1.0 second
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/runtime_status_test.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type | executor          | tags          | testpath
-----+-----+-----+-----+-----+-----
↳
↳
timelimit_min_max | bf6aaa60 | script | generic.local.sh | ['tutorials'] | /home/docs/
↳checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳local.sh/runtime_status_test/timelimit_min_max/bf6aaa60/timelimit_min_max_build.sh
timelimit_min      | 810cab9e | script | generic.local.sh | ['tutorials'] | /home/docs/
↳checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳local.sh/runtime_status_test/timelimit_min/810cab9e/timelimit_min_build.sh

```

(continued from previous page)

```

timelimit_max      | 8385a7e5 | script | generic.local.sh | ['tutorials'] | /home/docs/
↳checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳local.sh/runtime_status_test/timelimit_max/8385a7e5/timelimit_max_build.sh
timelimit_min_fail | 5352acc1 | script | generic.local.sh | ['tutorials'] | /home/docs/
↳checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳local.sh/runtime_status_test/timelimit_min_fail/5352acc1/timelimit_min_fail_build.sh
timelimit_max_fail | 40bee7c5 | script | generic.local.sh | ['tutorials'] | /home/docs/
↳checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳local.sh/runtime_status_test/timelimit_max_fail/40bee7c5/timelimit_max_fail_build.sh

+-----+
| Stage: Running Test |
+-----+

timelimit_min_max/bf6aaa60: completed with returncode: 0
timelimit_min_max/bf6aaa60: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_min_max/bf6aaa60/timelimit_min_max.out
timelimit_min_max/bf6aaa60: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_min_max/bf6aaa60/timelimit_min_max.err
timelimit_min_fail/5352acc1: completed with returncode: 0
timelimit_min_fail/5352acc1: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_min_fail/5352acc1/timelimit_min_fail.out
timelimit_min_fail/5352acc1: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_min_fail/5352acc1/timelimit_min_fail.err
timelimit_min/810cab9e: completed with returncode: 0
timelimit_min/810cab9e: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_min/810cab9e/timelimit_min.out
timelimit_min/810cab9e: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_min/810cab9e/timelimit_min.err
timelimit_max/8385a7e5: completed with returncode: 0
timelimit_max/8385a7e5: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_max/8385a7e5/timelimit_max.out
timelimit_max/8385a7e5: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_max/8385a7e5/timelimit_max.err
timelimit_max_fail/40bee7c5: completed with returncode: 0
timelimit_max_fail/40bee7c5: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_max_fail/40bee7c5/timelimit_max_fail.out
timelimit_max_fail/40bee7c5: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/runtime_status_test/
↳timelimit_max_fail/40bee7c5/timelimit_max_fail.err

-----
Launching test: timelimit_min_max
Test ID: bf6aaa60-4a8b-4249-9637-56b687aa201f

```

(continues on next page)

(continued from previous page)

```

Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/runtime_status_test/timelimit_min_max/bf6aaa60/
↳timelimit_min_max_build.sh

```

```

-----
Launching test: timelimit_min
Test ID: 810cab9e-06db-42d8-8516-3a56fa1b97cb
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/runtime_status_test/timelimit_min/810cab9e/timelimit_
↳min_build.sh

```

```

-----
Launching test: timelimit_max
Test ID: 8385a7e5-a8d8-4c1e-8999-ab8ebfb7430c
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/runtime_status_test/timelimit_max/8385a7e5/timelimit_
↳max_build.sh

```

```

-----
Launching test: timelimit_min_fail
Test ID: 5352acc1-0dbc-44ba-985f-a7a9d585974b
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/runtime_status_test/timelimit_min_fail/5352acc1/
↳timelimit_min_fail_build.sh

```

```

-----
Launching test: timelimit_max_fail
Test ID: 40bee7c5-9879-448b-a5e2-e055cf41f5fd
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/runtime_status_test/timelimit_max_fail/40bee7c5/
↳timelimit_max_fail_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_match	
↳regex_match	↳runtime_match	↳returncode	↳runtime		

↳timelimit_min_max	bf6aaa60	generic.local.sh	PASS	False	↳
↳False	True	0	2.04555		
↳timelimit_min	810cab9e	generic.local.sh	PASS	False	↳
↳False	True	0	2.03131		
↳timelimit_max	8385a7e5	generic.local.sh	PASS	False	↳
↳False	True	0	2.01983		
↳timelimit_min_fail	5352acc1	generic.local.sh	FAIL	False	↳
↳False	False	0	2.03991		
↳timelimit_max_fail	40bee7c5	generic.local.sh	FAIL	False	↳
↳False	False	0	3.03514		

(continues on next page)

(continued from previous page)

```
Passed Tests: 3/5 Percentage: 60.000%
Failed Tests: 2/5 Percentage: 40.000%
```

```
Writing Logfile to: /tmp/buildtest_qswjdps6.log
```

```
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log
```

If we look at the test results, we expect the first three tests **timelimit_min**, **timelimit_max**, **timelimit_min_max** will pass while the last two tests fail because it fails to comply with runtime property.

```
$ buildtest report --filter buildspectutorialruntime_status_test.yml --format name,id,
↳ state,runtime --latest
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/report.json
```

name	id	state	runtime
timelimit_min_max	bf6aaa60	PASS	2.04555
timelimit_min	810cab9e	PASS	2.03131
timelimit_max	8385a7e5	PASS	2.01983
timelimit_min_fail	5352acc1	FAIL	2.03991
timelimit_max_fail	40bee7c5	FAIL	3.03514

Defining Tags

The tags field can be used to classify tests which can be used to organize tests or if you want to *Building By Tags* (buildtest build --tags <TAGNAME>). Tags can be defined as a string or list of strings. In this example, the test string_tag defines a tag name **network** while test list_of_strings_tags define a list of tags named **network** and **ping**.

```
version: "1.0"
buildspecs:
  string_tag:
    type: script
    executor: generic.local.bash
    description: tags can be a string
    tags: network
    run: hostname

  list_of_strings_tags:
    type: script
    executor: generic.local.bash
```

(continues on next page)

(continued from previous page)

```

description: tags can be a list of strings
tags: [network, ping]
run: ping -c 4 www.google.com

```

Each item in tags must be a string and no duplicates are allowed, for example in this test, we define a duplicate tag **network** which is not allowed.

```

version: "1.0"
buildspecs:
  duplicate_string_tags:
    type: script
    executor: generic.local.bash
    description: duplicate strings in tags list is not allowed
    tags: [network, network]
    run: hostname

```

If we run this test and inspect the logs we will see an error message in schema validation:

```

2020-09-29 10:56:43,175 [parser.py:179 - _validate() ] - [INFO] Validating test -
↳ 'duplicate_string_tags' with schemafile: script-v1.0.schema.json
2020-09-29 10:56:43,175 [buildspec.py:397 - parse_buildspecs() ] - [ERROR] ['network',
↳ 'network'] is not valid under any of the given schemas

Failed validating 'oneOf' in schema['properties']['tags']:
  {'oneOf': [{'type': 'string'},
              {'$ref': '#/definitions/list_of_strings'}]}

On instance['tags']:
  ['network', 'network']

```

If tags is a list, it must contain one item, therefore an empty list (i.e tags: `[]`) is invalid.

Customize Shell

Shell Type

buildtest will default to bash shell when running test, but we can configure shell option using the `shell` field. The `shell` field is defined in schema as follows:

```

"shell": {
  "type": "string",
  "description": "Specify a shell launcher to use when running jobs. This sets the
↳ shebang line in your test script. The ``shell`` key can be used with ``run`` section
↳ to describe content of script and how its executed",
  "pattern": "^(/bin/bash|/bin/sh|/bin/csh|/bin/tcsh|/bin/
↳ zsh|bash|sh|csh|tcsh|zsh|python).*"
},

```

The shell pattern is a regular expression where one can specify a shell name along with shell options. The shell will configure the `shebang` in the test-script. In this example, we illustrate a few tests using different shell field.

```

version: "1.0"
buildspecs:
  _bin_sh_shell:
    executor: generic.local.sh
    type: script
    description: "/bin/sh shell example"
    shell: /bin/sh
    tags: [tutorials]
    run: "bzip2 --help"

  _bin_bash_shell:
    executor: generic.local.bash
    type: script
    description: "/bin/bash shell example"
    shell: /bin/bash
    tags: [tutorials]
    run: "bzip2 -h"

  bash_shell:
    executor: generic.local.bash
    type: script
    description: "bash shell example"
    shell: bash
    tags: [tutorials]
    run: "echo $SHELL"

  sh_shell:
    executor: generic.local.sh
    type: script
    description: "sh shell example"
    shell: sh
    tags: [tutorials]
    run: "echo $SHELL"

  shell_options:
    executor: generic.local.sh
    type: script
    description: "shell options"
    shell: "sh -x"
    tags: [tutorials]
    run: |
      echo $SHELL
      hostname

```

The generated test-script for buildspec `_bin_sh_shell` will specify shebang `/bin/sh` because we specified `shell: /bin/sh`:

```

#!/bin/sh
# Content of run section
bzip2 --help

```

If you don't specify a shell path such as `shell: sh`, then buildtest will resolve path by looking in `$PATH` and build the shebang line.

In test **shell_options** we specify shell: "sh -x", buildtest will tack on the shell options into the called script as follows:

```
#!/bin/bash

##### START VARIABLE DECLARATION #####
export BUILDTEST_TEST_NAME=shell_options
export BUILDTEST_TEST_ROOT=/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↳generic.local.sh/shell_examples/shell_options/0
export BUILDTEST_BUILDSPEC_DIR=/Users/siddiq90/Documents/GitHubDesktop/buildtest/
↳tutorials
export BUILDTEST_STAGE_DIR=/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↳generic.local.sh/shell_examples/shell_options/0/stage
export BUILDTEST_TEST_ID=95c11f54-bbb1-4154-849d-44313e4417c2
##### END VARIABLE DECLARATION #####

# source executor startup script
source /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/executor/generic.local.sh/
↳before_script.sh
# Run generated script
sh -x /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.sh/shell_
↳examples/shell_options/0/stage/shell_options.sh
# Get return code
returncode=$?
# Exit with return code
exit $returncode
```

If you prefer **cs**h or **tc**sh for writing scripts just set shell: csh or shell: tcsh, note you will need to match this with appropriate executor. For now use executor: generic.local.csh to run your csh/tcsh scripts. In this example below we define a script using csh, take note of run section we can write csh style.

```
version: "1.0"
buildspecs:
  csh_shell:
    executor: generic.local.csh
    type: script
    description: "csh shell example"
    shell: csh
    tags: [tutorials]
    vars:
      file: "/etc/csh.cshrc"
    run: |
      if (-e $file) then
        echo "$file file found"
      else
        echo "$file file not found"
        exit 1
      endif
```


Customize Shebang

You may customize the shebang line in testscript using `shebang` field. This takes precedence over the `shell` property which automatically detects the shebang based on shell path.

In next example we have two tests `bash_login_shebang` and `bash_nonlogin_shebang` which tests if shell is Login or Non-Login. The `#!/bin/bash -l` indicates we want to run in login shell and expects an output of `Login Shell` while test `bash_nonlogin_shebang` should run in default behavior which is non-login shell and expects output `Not Login Shell`. We match this with regular expression with stdout stream.

```
version: "1.0"
buildspecs:
  bash_login_shebang:
    type: script
    executor: generic.local.bash
    shebang: "#!/bin/bash -l"
    description: customize shebang line with bash login shell
    tags: tutorials
    run: shopt -q login_shell && echo 'Login Shell' || echo 'Not Login Shell'
    status:
      regex:
        exp: "^Login Shell$"
        stream: stdout

  bash_nonlogin_shebang:
    type: script
    executor: generic.local.bash
    shebang: "#!/bin/bash"
    description: customize shebang line with default bash (nonlogin) shell
    tags: tutorials
    run: shopt -q login_shell && echo 'Login Shell' || echo 'Not Login Shell'
    status:
      regex:
        exp: "^Not Login Shell$"
        stream: stdout
```

Now let's run this test as we see the following.

```
$ buildtest build -b tutorials/shebang.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:54:59
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/shebang.yml
```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪ -----+
| Discovered Buildsspecs                                     ↪
↪           |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ shebang.yml |
+-----+
↪ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ shebang.yml: VALID

Total builder objects created: 2
builders: [bash_login_shebang/3415a842, bash_nonlogin_shebang/52864b0a]

name           id           description                                     ↪
↪ buildsspecs
-----
↪ - -----
↪ -----
bash_login_shebang 3415a842 customize shebang line with bash login shell
↪ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪ tutorials/shebang.yml
bash_nonlogin_shebang 52864b0a customize shebang line with default bash (nonlogin) ↪
↪ shell /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪ tutorials/shebang.yml

+-----+
| Stage: Building Test |
+-----+

name           | id           | type   | executor           | tags           | testpath
-----+-----+-----+-----+-----+-----
↪ -----
↪ -----

```

(continues on next page)

(continued from previous page)

```

bash_login_shebang | 3415a842 | script | generic.local.bash | tutorials | /home/docs/
↳ checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳ local.bash/shebang/bash_login_shebang/3415a842/bash_login_shebang_build.sh
bash_nonlogin_shebang | 52864b0a | script | generic.local.bash | tutorials | /home/docs/
↳ checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳ local.bash/shebang/bash_nonlogin_shebang/52864b0a/bash_nonlogin_shebang_build.sh

+-----+
| Stage: Running Test |
+-----+

bash_nonlogin_shebang/52864b0a: completed with returncode: 0
bash_nonlogin_shebang/52864b0a: Writing output file: /home/docs/checkouts/readthedocs.
↳ org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/shebang/bash_
↳ nonlogin_shebang/52864b0a/bash_nonlogin_shebang.out
bash_nonlogin_shebang/52864b0a: Writing error file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/shebang/bash_
↳ nonlogin_shebang/52864b0a/bash_nonlogin_shebang.err
bash_nonlogin_shebang/52864b0a: performing regular expression - '^Not Login Shell$' on
↳ file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.bash/shebang/bash_nonlogin_shebang/52864b0a/bash_nonlogin_shebang.
↳ out with regular expression
bash_nonlogin_shebang/52864b0a: Regular Expression Match - Success!
bash_login_shebang/3415a842: completed with returncode: 0
bash_login_shebang/3415a842: Writing output file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/shebang/bash_
↳ login_shebang/3415a842/bash_login_shebang.out
bash_login_shebang/3415a842: Writing error file: /home/docs/checkouts/readthedocs.org/
↳ user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/shebang/bash_
↳ login_shebang/3415a842/bash_login_shebang.err
bash_login_shebang/3415a842: performing regular expression - '^Login Shell$' on file: /
↳ home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳ generic.local.bash/shebang/bash_login_shebang/3415a842/bash_login_shebang.out with
↳ regular expression
bash_login_shebang/3415a842: Regular Expression Match - Success!

-----
Launching test: bash_login_shebang
Test ID: 3415a842-ed4b-435d-a46f-7f942694f28b
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/shebang/bash_login_shebang/3415a842/bash_login_
↳ shebang_build.sh

-----
Launching test: bash_nonlogin_shebang
Test ID: 52864b0a-dc87-4e8e-a8df-ba8f8b7f98ff
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/shebang/bash_nonlogin_shebang/52864b0a/bash_nonlogin_
↳ shebang_build.sh

+-----+
| Stage: Test Summary |
+-----+

```

(continues on next page)

(continued from previous page)

+-----+					
name	id	executor	status	returncode_match	
↪ regex_match	runtime_match	returncode	runtime		
+-----+					
↪ bash_login_shebang	3415a842	generic.local.bash	PASS	False	↪
↪ True	False	0	0.030688		
↪ bash_nonlogin_shebang	52864b0a	generic.local.bash	PASS	False	↪
↪ True	False	0	0.022097		

Passed Tests: 2/2 Percentage: 100.000%
Failed Tests: 0/2 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_1n3al6un.log
A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↪ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

If we look at the generated test for **bash_login_shebang** we see the shebang line is passed into the script:

```
#!/bin/bash -l
# Content of run section
shopt -q login_shell && echo 'Login Shell' || echo 'Not Login Shell'
```

Python Shell

You can use **script** schema to write python scripts using the **run** property. This can be achieved if you use the **generic.local.python** executor assuming you have this defined in your buildtest configuration.

Here is a python example calculating area of circle

```
version: "1.0"
buildspecs:
  circle_area:
    executor: generic.local.python
    type: script
    shell: python
    description: "Calculate circle of area given a radius"
    tags: [tutorials, python]
    run: |
      import math
      radius = 2
      area = math.pi * radius * radius
      print("Circle Radius ", radius)
      print("Area of circle ", area)
```

The **shell: python** will let us write python script in the **run** section. The **tags** field can be used to classify test, the field expects an array of string items.

Note: Python scripts are very picky when it comes to formatting, in the `run` section if you are defining multiline python script you must remember to use 2 space indent to register multiline string. buildtest will extract the content from run section and inject in your test script. To ensure proper formatting for a more complex python script you may be better off writing a python script in separate file and call it in `run` section.

Skipping test

By default, buildtest will run all tests defined in `buildspecs` section, if you want to skip a test use the `skip` field which expects a boolean value. Shown below is an example test.

```
version: "1.0"
buildspecs:
  skip:
    type: script
    executor: generic.local.bash
    description: This test is skipped
    skip: Yes
    tags: [tutorials]
    run: hostname

  unskipped:
    type: script
    executor: generic.local.bash
    description: This test is not skipped
    skip: No
    tags: [tutorials]
    run: hostname
```

The first test **skip** will be ignored by buildtest because `skip: true` is defined while **unskipped** will be processed as usual.

Note: YAML and JSON have different representation for boolean. For json schema valid values are `true` and `false` see <https://json-schema.org/understanding-json-schema/reference/boolean.html> however YAML has many more representation for boolean see <https://yaml.org/type/bool.html>. You may use any of the YAML boolean, however it's best to stick with json schema values `true` and `false`.

Here is an example build, notice message `[skip] test is skipped` during the build stage

```
$ buildtest build -b tutorials/skip_tests.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:00
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
```

(continues on next page)

(continued from previous page)

```

Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳bin/buildtest build -b tutorials/skip_tests.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳-----+
| Discovered Buildsspecs |
↳      |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳skip_tests.yml |
+-----+
↳-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

[skip](/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳tutorials/skip_tests.yml): test is skipped.
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳skip_tests.yml: VALID

Total builder objects created: 1
builders: [unskipped/27e1bc21]

name      id      description      buildsspecs
-----
↳-----
unskipped 27e1bc21 This test is not skipped /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/tutorials/skip_tests.yml

+-----+
| Stage: Building Test |
+-----+

name      | id      | type  | executor      | tags      | testpath
-----+-----+-----+-----+-----+-----
↳-----
↳-----

```

(continues on next page)

(continued from previous page)

```

unskipped | 27e1bc21 | script | generic.local.bash | ['tutorials'] | /home/docs/
↳ checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↳ local.bash/skip_tests/unskipped/27e1bc21/unskipped_build.sh

+-----+
| Stage: Running Test |
+-----+

unskipped/27e1bc21: completed with returncode: 0
unskipped/27e1bc21: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/skip_tests/unskipped/
↳ 27e1bc21/unskipped.out
unskipped/27e1bc21: Writing error file: /home/docs/checkouts/readthedocs.org/user_builds/
↳ buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/skip_tests/unskipped/27e1bc21/
↳ unskipped.err

-----
Launching test: unskipped
Test ID: 27e1bc21-7820-4c39-aa31-1b91e1e27760
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/skip_tests/unskipped/27e1bc21/unskipped_build.sh

+-----+
| Stage: Test Summary |
+-----+

name      | id      | executor      | status | returncode_match | regex_match_
↳ | runtime_match | returncode | runtime
-----+-----+-----+-----+-----+-----+
↳ +-----+-----+-----+-----+-----+-----+
unskipped | 27e1bc21 | generic.local.bash | PASS   | N/A              | N/A
↳ | N/A      | 0 | 0.006467

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_eng8vf0m.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

```

Defining Metrics

buildtest provides a method to define test metrics in the buildsspecs which can be used to store arbitrary content from the output/error file into named metric. A metric is defined using the `metrics` property where each element under the `metrics` property is the name of the metric which must be a unique name. A metric can apply regular expression on stdout, stderr like in this example below. The metrics are captured in the test report which can be queried via `buildtest report` or `buildtest inspect`. Shown below is an example where we define two metrics named `hpcg_rating` and `hpcg_state`.

```
version: "1.0"
buildspecs:
  metric_regex_example:
    executor: generic.local.sh
    type: script
    description: capture result metric from output
    run: echo "HPCG result is VALID with a GFLOP/s rating of=63.6515"
    tags: tutorials
  metrics:
    hpcg_rating:
      regex:
        exp: 'rating of=(\d+\.\d+)$'
        stream: stdout

    hpcg_state:
      regex:
        exp: '(VALID)'
        stream: stdout
```

The metrics will not impact behavior of test, it will only impact the test report. By default a metric will be an empty dictionary if there is no `metrics` property. If we fail to match a regular expression, the metric will be defined as an empty string.

Note: If your regular expression contains an escape character `\` you must surround your string in single quotes `'` as pose to double quotes `"`

Let's build this test.

```
$ buildtest build -b tutorials/metrics_regex.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:00
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/metrics_regex.yml
```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪ -----+
| Discovered Buildsspecs                                     ↪
↪ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ metrics_regex.yml |
+-----+
↪ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ metrics_regex.yml: VALID

Total builder objects created: 1
builders: [metric_regex_example/15948b76]

name            id            description                                buildsspecs
-----
↪ -----
metric_regex_example 15948b76 capture result metric from output /home/docs/checkouts/
↪ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/metrics_regex.yml

+-----+
| Stage: Building Test |
+-----+

name            | id            | type  | executor          | tags      | testpath
-----+-----+-----+-----+-----+-----
↪ -----
↪ -----
metric_regex_example | 15948b76 | script | generic.local.sh | tutorials | /home/docs/
↪ checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.
↪ local.sh/metrics_regex/metric_regex_example/15948b76/metric_regex_example_build.sh

+-----+
| Stage: Running Test |

```

(continues on next page)

(continued from previous page)

```

+-----+
metric_regex_example/15948b76: completed with returncode: 0
metric_regex_example/15948b76: Writing output file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/metrics_regex/
↳metric_regex_example/15948b76/metric_regex_example.out
metric_regex_example/15948b76: Writing error file: /home/docs/checkouts/readthedocs.org/
↳user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/metrics_regex/
↳metric_regex_example/15948b76/metric_regex_example.err

Launching test: metric_regex_example
Test ID: 15948b76-7261-4149-8581-39fa11dfdec9
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/metrics_regex/metric_regex_example/15948b76/metric_
↳regex_example_build.sh

+-----+
| Stage: Test Summary |
+-----+

name          | id          | executor          | status | returncode_match |
↳regex_match  | runtime_match | returncode | runtime
+-----+-----+-----+-----+-----+-----+
↳-----+-----+-----+-----+-----+-----+
metric_regex_example | 15948b76 | generic.local.sh | PASS   | N/A              | N/
↳A              | N/A              | 0 | 0.005627

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_p5lwgqj9.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

```

We can query the metrics via `buildtest report` which will display all metrics as a comma separated **Key/Value** pair. We can use `buildtest report --format metrics` to extract all metrics for a test. Internally, we store the metrics as a dictionary but when we print them out via `buildtest report` we join them together into a single string. Shown below is the metrics for the previous build.

```

$ buildtest report --filter buildspectutorialsmetrics_regex.yml --format name,metrics
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/report.json

+-----+-----+
| name          | metrics          |
+=====+=====+
| metric_regex_example | hpcg_rating=rating of=63.6515,hpcg_state=VALID |
+-----+-----+

```

You can define a metric based on *variables* or *environment variables* which requires you have set vars or env property in the builds spec. The vars and env is a property under the metric name that can be used to reference name of variable or environment variable. If you reference an invalid name, buildtest will assign the metric an empty string. In this next example, we define two metrics gflop and foo that are assigned to variable GFLOPS and environment variable FOO.

```
version: "1.0"
buildspecs:
  metric_variable_assignment:
    executor: generic.local.sh
    type: script
    description: capture result metric based on variables and environment variable
    vars:
      GFLOPS: "63.6515"
    env:
      FOO: BAR
    run: |
      echo $GFLOPS
      echo $FOO
    tags: tutorials
  metrics:
    gflops:
      vars: "GFLOPS"
    foo:
      env: "FOO"
```

Now let's build the test.

```
$ buildtest build -b tutorials/metrics_variable.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:01
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/metrics_variable.yml
```

```
+-----+
| Stage: Discovering Buildsspecs |
+-----+
```

```
+-----+
↳ -----+
| Discovered Buildsspecs
↳ |
+=====
```

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_variable.yml |
+-----+
↳-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳metrics_variable.yml: VALID

Total builder objects created: 1
builders: [metric_variable_assignment/0fbf904d]

name                id                description
↳                buildsspecs
-----
↳-----
↳-----
metric_variable_assignment 0fbf904d capture result metric based on variables and
↳environment variable /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/tutorials/metrics_variable.yml

+-----+
| Stage: Building Test |
+-----+

name                | id                | type  | executor          | tags          | testpath
-----+-----+-----+-----+-----+-----
↳-----
↳-----
↳-----
metric_variable_assignment | 0fbf904d | script | generic.local.sh | tutorials | /home/
↳docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.sh/metrics_variable/metric_variable_assignment/0fbf904d/metric_variable_
↳assignment_build.sh

+-----+
| Stage: Running Test |
+-----+

metric_variable_assignment/0fbf904d: completed with returncode: 0
metric_variable_assignment/0fbf904d: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳metrics_variable/metric_variable_assignment/0fbf904d/metric_variable_assignment.out
↳(next page)
```

(continued from previous page)

```
metric_variable_assignment/0fbf904d: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳metrics_variable/metric_variable_assignment/0fbf904d/metric_variable_assignment.err
```

```
Launching test: metric_variable_assignment
Test ID: 0fbf904d-e32a-429e-9a83-8007ac59ef08
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/metrics_variable/metric_variable_assignment/0fbf904d/
↳metric_variable_assignment_build.sh
```

```
+-----+
| Stage: Test Summary |
+-----+
```

name	id	executor	status	returncode_match
↳ regex_match	↳ runtime_match	↳ returncode	↳ runtime	
metric_variable_assignment	0fbf904d	generic.local.sh	PASS	N/A
↳ N/A	↳ N/A	↳ 0	↳ 0.005851	

```
Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%
```

```
Writing Logfile to: /tmp/buildtest_1_5x3yi0.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log
```

Now if we query the previous test, we will see the two metrics gflops and foo are captured in the test.

```
$ buildtest report --filter buildspectutorialsmetrics_variable.yml --format name,
↳metrics
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/var/report.json
```

```
+-----+
| name | metrics |
+-----+
| metric_variable_assignment | gflops=63.6515,foo=BAR |
+-----+
```

You can also define metrics with the *compiler schema* which works slightly different when it comes to variable and environment assignment. Since you can define vars and env in defaults or config section. Let's take a look at this next example where we compile an openmp code that will use the *OMP_NUM_THREADS* environment as the metric that is assigned to name openmp_threads. Since we have defined OMP_NUM_THREADS under the defaults and config section we will use the environment variable that corresponds to each compiler.

version: "1.0"

(continues on next page)

(continued from previous page)

```

buildspecs:
  metrics_variable_compiler:
    type: compiler
    description: define metrics with compiler schema
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/hello_omp.c"
    compilers:
      name: ["^(builtin_gcc|gcc)"]
      default:
        gcc:
          cflags: -fopenmp
          env:
            OMP_NUM_THREADS: 4
        config:
          builtin_gcc:
            env:
              OMP_NUM_THREADS: 1
          gcc/9.3.0-n7p74fd:
            env:
              OMP_NUM_THREADS: 2

    metrics:
      openmp_threads:
        env: "OMP_NUM_THREADS"

```

Note: This test uses a custom site configuration that defines gcc multiple compilers.

Let's build this test as follows

```

$ buildtest -c config/laptop.yml build -b tutorials/compilers/metrics_openmp.yml

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/07/24 00:14:33
buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.10.0
python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
Configuration File: /Users/siddiq90/Documents/GitHubDesktop/buildtest/config/laptop.yml
Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest -c config/
↳ laptop.yml build -b tutorials/compilers/metrics_openmp.yml

```

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

```

```

+-----+
↳ --+

```

(continues on next page)

(continued from previous page)

```

| Discovered Buildsspecs
↪ |
=====+
| /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compilers/metrics_openmp.
↪ yml |
+-----+
↪ --+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

schemafile          | validstate | buildspec
+-----+-----+-----+
↪ -----+
compiler-v1.0.schema.json | True       | /Users/siddiq90/Documents/GitHubDesktop/
↪ buildtest/tutorials/compilers/metrics_openmp.yml

name                  description
+-----+-----+
metrics_variable_compiler | define metrics with compiler schema
metrics_variable_compiler | define metrics with compiler schema
metrics_variable_compiler | define metrics with compiler schema

+-----+
| Stage: Building Test |
+-----+

name                  | id          | type      | executor          | tags
↪ | compiler          | testpath
+-----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
metrics_variable_compiler | e45976b8 | compiler | generic.local.bash | ['tutorials',
↪ 'compile'] | builtin_gcc | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↪ var/tests/generic.local.bash/metrics_openmp/metrics_variable_compiler/11/metrics_
↪ variable_compiler_build.sh
metrics_variable_compiler | 8bc71f19 | compiler | generic.local.bash | ['tutorials',
↪ 'compile'] | gcc/9.3.0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↪ var/tests/generic.local.bash/metrics_openmp/metrics_variable_compiler/12/metrics_
↪ variable_compiler_build.sh
metrics_variable_compiler | 7127eb46 | compiler | generic.local.bash | ['tutorials',
↪ 'compile'] | gcc/10.2.0-37fmsw7 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↪ var/tests/generic.local.bash/metrics_openmp/metrics_variable_compiler/13/metrics_
↪ variable_compiler_build.sh

```

(continued from previous page)

```

+-----+
| Stage: Running Test |
+-----+

name          | id          | executor          | status  | returncode
+-----+
metrics_variable_compiler | e45976b8 | generic.local.bash | FAIL    | 127
metrics_variable_compiler | 8bc71f19 | generic.local.bash | PASS    | 0
metrics_variable_compiler | 7127eb46 | generic.local.bash | PASS    | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 2/3 Percentage: 66.667%
Failed Tests: 1/3 Percentage: 33.333%

Writing Logfile to: /Users/siddiq90/buildtest/buildtest_0a04808e.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
↳ Documents/GitHubDesktop/buildtest/buildtest.log

```

Now if we filter the results, notice that `builtin_gcc` got metrics `openmp_threads=1` since that is the value set under the `builtin_gcc` compiler instance under the config section. The `gcc/9.3.0-n7p74fd` compiler got value of **2** because we have an entry defined under the config section while `gcc/10.2.0-37fmsw7` compiler got the value of **4** from the default section that is inherited for all gcc compilers.

```

$ buildtest report --filter buildspectutorials/compilers/metrics_openmp.yml --format_
↳ name,compiler,metrics
Reading report file: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/report.json

```

```

+-----+-----+-----+
| name          | compiler          | metrics          |
+-----+-----+-----+
| metrics_variable_compiler | builtin_gcc      | openmp_threads=1 |
+-----+-----+-----+
| metrics_variable_compiler | gcc/9.3.0-n7p74fd | openmp_threads=2 |
+-----+-----+-----+
| metrics_variable_compiler | gcc/10.2.0-37fmsw7 | openmp_threads=4 |
+-----+-----+-----+

```


Running test across multiple executors

The *executor* property can support regular expression to search for compatible executors, this can be used if you want to run a test across multiple executors. In buildtest, we use `re.fullmatch` with the input pattern defined by **executor** property against a list of available executors defined in configuration file. You can retrieve a list of executors by running `buildtest config executors`.

In example below we will run this test on *generic.local.bash* and *generic.local.sh* executor based on the regular expression.

```
version: "1.0"
buildspecs:
  executor_regex_script_schema:
    type: script
    executor: 'generic.local.(bash|sh)'
    description: regular expression test with executor using script schema
    tags: [tutorials]
    run: date
```

If we build this test, notice that there are two tests, one for each executor.

```
$ buildtest build -b tutorials/executor_regex_script.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:01
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/executor_regex_script.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ +-----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ executor_regex_script.yml |
+-----+
↳ +-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
```

(continues on next page)

(continued from previous page)

Detected Buildsspecs after exclusion: 1

```

+-----+
| Stage: Parsing Buildsspecs |
+-----+

```

Valid Buildsspecs: 1

Invalid Buildsspecs: 0

```

/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ executor_regex_script.yml: VALID

```

Total builder objects created: 2

builders: [executor_regex_script_schema/1bc9d0da, executor_regex_script_schema/4220359a]

name	id	description
↳ buildsspecs		

↳		

↳ executor_regex_script_schema	1bc9d0da	regular expression test with executor using
↳ script schema	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/executor_regex_script.yml	
↳ executor_regex_script_schema	4220359a	regular expression test with executor using
↳ script schema	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/executor_regex_script.yml	

```

+-----+
| Stage: Building Test |
+-----+

```

name	id	type	executor	tags
↳ testpath				

↳				

↳ executor_regex_script_schema	1bc9d0da	script	generic.local.bash	['tutorials']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/executor_regex_script/executor_regex_script_schema/1bc9d0da/executor_regex_script_schema_build.sh				
↳ executor_regex_script_schema	4220359a	script	generic.local.sh	['tutorials']
↳ /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/executor_regex_script/executor_regex_script_schema/4220359a/executor_regex_script_schema_build.sh				

```

+-----+
| Stage: Running Test |
+-----+

```

executor_regex_script_schema/4220359a: completed with returncode: 0

(continues on next page)

(continued from previous page)

```

executor_regex_script_schema/4220359a: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳ executor_regex_script/executor_regex_script_schema/4220359a/executor_regex_script_
↳ schema.out
executor_regex_script_schema/4220359a: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳ executor_regex_script/executor_regex_script_schema/4220359a/executor_regex_script_
↳ schema.err
executor_regex_script_schema/1bc9d0da: completed with returncode: 0
executor_regex_script_schema/1bc9d0da: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ executor_regex_script/executor_regex_script_schema/1bc9d0da/executor_regex_script_
↳ schema.out
executor_regex_script_schema/1bc9d0da: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ executor_regex_script/executor_regex_script_schema/1bc9d0da/executor_regex_script_
↳ schema.err

```

```

-----
Launching test: executor_regex_script_schema
Test ID: 1bc9d0da-7ec3-4131-ad36-338cf98d229e
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/executor_regex_script/executor_regex_script_schema/
↳ 1bc9d0da/executor_regex_script_schema_build.sh

```

```

-----
Launching test: executor_regex_script_schema
Test ID: 4220359a-940f-4aab-890f-29bab98ebc2b
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/executor_regex_script/executor_regex_script_schema/
↳ 4220359a/executor_regex_script_schema_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_
↳ match	↳ regex_match	↳ runtime_match	↳ returncode	↳ runtime
executor_regex_script_schema	1bc9d0da	generic.local.bash	PASS	N/A
↳	↳ N/A	↳ N/A	↳ 0	↳ 0.032167
executor_regex_script_schema	4220359a	generic.local.sh	PASS	N/A
↳	↳ N/A	↳ N/A	↳ 0	↳ 0.026747

```

Passed Tests: 2/2 Percentage: 100.000%
Failed Tests: 0/2 Percentage: 0.000%

```

```

Writing Logfile to: /tmp/buildtest_fvx48ab1.log

```

(continues on next page)

(continued from previous page)

A copy of logfile can be found at \$BUILDTTEST_ROOT/buildtest.log - /home/docs/checkouts/
readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

Multiple Executors

Note: This feature is in active development

Note: This feature is compatible with `type: script` and `type: spack`.

The `executors` property can be used to define executor specific configuration for each test, currently this field can be used with `vars`, `env`, scheduler directives: `sbatch`, `bsub`, `pbs`, `cobalt` and `cray burst buffer/data warp`. The `executors` field is a JSON object that expects name of executor followed by property set per executor. In this next example, we define variables X, Y and environment SHELL based on executors `generic.local.sh` and `generic.local.bash`.

```
version: "1.0"
buildspecs:
  executors_vars_env_declaration:
    type: script
    executor: 'generic.local.(bash|sh)'
    description: Declaring env and vars by executors section
    tags: [tutorials]
    run: |
      echo "X:" $X
      echo "Y:" $Y
      echo $SHELL

  executors:
    generic.local.bash:
      vars:
        X: 1
        Y: 3
      env:
        SHELL: bash
    generic.local.sh:
      vars:
        X: 2
        Y: 4
      env:
        SHELL: sh
```

Let's build this test.

```
$ buildtest build -b tutorials/script/multiple_executors.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:02
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
11.0/bin/buildtest
```

(continues on next page)

(continued from previous page)

```

buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳bin/buildtest build -b tutorials/script/multiple_executors.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳-----+
| Discovered Buildsspecs |
↳
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml |
+-----+
↳-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml: VALID

Total builder objects created: 2
builders: [executors_vars_env_declaration/cc02b315, executors_vars_env_declaration/
↳b19d2068]

name            id            description
↳buildspecs
-----
↳-----
executors_vars_env_declaration cc02b315 Declaring env and vars by executors section /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml
executors_vars_env_declaration b19d2068 Declaring env and vars by executors section /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/multiple_executors.yml

```

(continued from previous page)

```

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags          |
+-----+-----+-----+-----+-----+
↳ | testpath
+-----+-----+-----+-----+-----+
↳ +-----+-----+-----+-----+-----+
↳ +-----+-----+-----+-----+-----+
↳ +-----+-----+-----+-----+-----+
executors_vars_env_declaration | cc02b315 | script | generic.local.bash | ['tutorials']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.bash/multiple_executors/executors_vars_env_declaration/cc02b315/
↳ executors_vars_env_declaration_build.sh
executors_vars_env_declaration | b19d2068 | script | generic.local.sh   | ['tutorials']
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/multiple_executors/executors_vars_env_declaration/b19d2068/
↳ executors_vars_env_declaration_build.sh

+-----+
| Stage: Running Test |
+-----+

executors_vars_env_declaration/cc02b315: completed with returncode: 0
executors_vars_env_declaration/cc02b315: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ multiple_executors/executors_vars_env_declaration/cc02b315/executors_vars_env_
↳ declaration.out
executors_vars_env_declaration/cc02b315: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ multiple_executors/executors_vars_env_declaration/cc02b315/executors_vars_env_
↳ declaration.err
executors_vars_env_declaration/b19d2068: completed with returncode: 0
executors_vars_env_declaration/b19d2068: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳ multiple_executors/executors_vars_env_declaration/b19d2068/executors_vars_env_
↳ declaration.out
executors_vars_env_declaration/b19d2068: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳ multiple_executors/executors_vars_env_declaration/b19d2068/executors_vars_env_
↳ declaration.err

-----
Launching test: executors_vars_env_declaration
Test ID: cc02b315-0c77-4958-bb16-4d181319c4a4
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/multiple_executors/executors_vars_env_declaration/
↳ cc02b315/executors_vars_env_declaration_build.sh

-----
Launching test: executors_vars_env_declaration
Test ID: b19d2068-1076-448e-93f4-18dbd54cac49

```

(continues on next page)

(continued from previous page)

```
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/multiple_executors/executors_vars_env_declaration/
↳b19d2068/executors_vars_env_declaration_build.sh
```

```
+-----+
| Stage: Test Summary |
+-----+
```

name	id	executor	status	returncode_
↳match	↳regex_match	↳runtime_match	↳returncode	↳runtime
executors_vars_env_declaration	cc02b315	generic.local.bash	PASS	N/A
↳	↳N/A	↳	↳0	↳0.026931
executors_vars_env_declaration	b19d2068	generic.local.sh	PASS	N/A
↳	↳N/A	↳	↳0	↳0.024118

Passed Tests: 2/2 Percentage: 100.000%

Failed Tests: 0/2 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_ebbywpnl.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

Now let's look at the generated content of the test as follows. We will see that buildtest will set **X=1, Y=3** and **SHELL=bash** for generic.local.bash and **X=2, Y=4** and **SHELL=sh** for generic.local.sh

```
$ buildtest inspect query -d all -t executors_vars_env_declaration
----- executors_vars_env_declaration (ID: cc02b315-0c77-4958-
↳bb16-4d181319c4a4) -----
executor: generic.local.bash
description: Declaring env and vars by executors section
state: PASS
returncode: 0
runtime: 0.026931
starttime: 2021/09/09 15:55:02
endtime: 2021/09/09 15:55:02
***** Start of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/multiple_executors/
↳executors_vars_env_declaration/cc02b315/executors_vars_env_declaration.sh
↳*****
#!/bin/bash
# Declare environment variables
export SHELL=bash

# Declare environment variables
export X=1
```

(continues on next page)

(continued from previous page)

```

export Y=3

# Content of run section
echo "X:" $X
echo "Y:" $Y
echo $SHELL

***** End of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/multiple_executors/
↳ executors_vars_env_declaration/cc02b315/executors_vars_env_declaration.sh
↳ *****

----- executors_vars_env_declaration (ID: b19d2068-1076-448e-
↳ 93f4-18dbd54cac49) -----
executor: generic.local.sh
description: Declaring env and vars by executors section
state: PASS
returncode: 0
runtime: 0.024118
starttime: 2021/09/09 15:55:02
endtime: 2021/09/09 15:55:02
***** Start of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/multiple_executors/
↳ executors_vars_env_declaration/b19d2068/executors_vars_env_declaration.sh
↳ *****
#!/bin/bash
# Declare environment variables
export SHELL=sh

# Declare environment variables
export X=2
export Y=4

# Content of run section
echo "X:" $X
echo "Y:" $Y
echo $SHELL

***** End of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/multiple_executors/
↳ executors_vars_env_declaration/b19d2068/executors_vars_env_declaration.sh
↳ *****

```


Scheduler Directives

We can also define scheduler directives based on executor type, in this example we define sbatch property per executor type. Note that sbatch property in the executors section will override the sbatch property defined in the top-level file otherwise it will use the default.

```
version: "1.0"
buildspecs:
  executors_sbatch_declaration:
    type: script
    executor: 'generic.local.(bash|sh)'
    description: Declaring env and vars by executors section
    tags: [tutorials]
    run: hostname
    sbatch: ["-N 4"]
    executors:
      generic.local.bash:
        sbatch: ["-n 4", "-N 1", "-t 30"]
      generic.local.sh:
        sbatch: ["-n 8", "-N 1", "-t 60"]
```

```
$ buildtest build -b tutorials/script/executor_scheduler.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:02
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/script/executor_scheduler.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ script/executor_scheduler.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
```

(continues on next page)

(continued from previous page)

```

Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/executor_scheduler.yml: VALID

Total builder objects created: 2
builders: [executors_sbatch_declaration/17f960fc, executors_sbatch_declaration/17627bec]

name                id                description
↳buildspecs
-----
↳
↳
executors_sbatch_declaration 17f960fc Declaring env and vars by executors section /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/executor_scheduler.yml
executors_sbatch_declaration 17627bec Declaring env and vars by executors section /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳script/executor_scheduler.yml

+-----+
| Stage: Building Test |
+-----+

name                | id          | type   | executor                | tags          |
↳testpath
-----+-----+-----+-----+-----+
↳
↳
↳
executors_sbatch_declaration | 17f960fc | script | generic.local.bash | ['tutorials'] |
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/executor_scheduler/executors_sbatch_declaration/17f960fc/executors_
↳sbatch_declaration_build.sh
executors_sbatch_declaration | 17627bec | script | generic.local.sh   | ['tutorials'] |
↳/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.sh/executor_scheduler/executors_sbatch_declaration/17627bec/executors_
↳sbatch_declaration_build.sh

+-----+
| Stage: Running Test |
+-----+

```

(continues on next page)

(continued from previous page)

```

executors_sbatch_declaration/17f960fc: completed with returncode: 0
executors_sbatch_declaration/17f960fc: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳executor_scheduler/executors_sbatch_declaration/17f960fc/executors_sbatch_declaration.
↳out
executors_sbatch_declaration/17f960fc: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳executor_scheduler/executors_sbatch_declaration/17f960fc/executors_sbatch_declaration.
↳err
executors_sbatch_declaration/17627bec: completed with returncode: 0
executors_sbatch_declaration/17627bec: Writing output file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳executor_scheduler/executors_sbatch_declaration/17627bec/executors_sbatch_declaration.
↳out
executors_sbatch_declaration/17627bec: Writing error file: /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳executor_scheduler/executors_sbatch_declaration/17627bec/executors_sbatch_declaration.
↳err

```

```

-----
Launching test: executors_sbatch_declaration
Test ID: 17f960fc-0691-4997-bd7f-b420d899a406
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/executor_scheduler/executors_sbatch_declaration/
↳17f960fc/executors_sbatch_declaration_build.sh

```

```

-----
Launching test: executors_sbatch_declaration
Test ID: 17627bec-636c-42d2-a55b-0429bb94a7d7
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.sh/executor_scheduler/executors_sbatch_declaration/
↳17627bec/executors_sbatch_declaration_build.sh

```

```

+-----+
| Stage: Test Summary |
+-----+

```

name	id	executor	status	returncode_
↳match	↳regex_match	↳runtime_match	↳returncode	↳runtime
executors_sbatch_declaration	17f960fc	generic.local.bash	PASS	N/A
↳	N/A	N/A	0	0.023433
executors_sbatch_declaration	17627bec	generic.local.sh	PASS	N/A
↳	N/A	N/A	0	0.027592

Passed Tests: 2/2 Percentage: 100.000%

Failed Tests: 0/2 Percentage: 0.000%

(continues on next page)

(continued from previous page)

```
Writing Logfile to: /tmp/buildtest_5mjf75ja.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log
```

If we inspect this test, we will see each each test have different #SBATCH directives for each test based on the sbatch property defined in the executors field.

```
$ buildtest inspect query -d all -t executors_sbatch_declaration
----- executors_sbatch_declaration (ID: 17f960fc-0691-4997-bd7f-
↳ b420d899a406) -----
executor: generic.local.bash
description: Declaring env and vars by executors section
state: PASS
returncode: 0
runtime: 0.023433
starttime: 2021/09/09 15:55:02
endtime: 2021/09/09 15:55:02
***** Start of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/executor_scheduler/
↳ executors_sbatch_declaration/17f960fc/executors_sbatch_declaration.sh_
↳ *****
#!/bin/bash
##### START OF SCHEDULER DIRECTIVES #####
#SBATCH -n 4
#SBATCH -N 1
#SBATCH -t 30
#SBATCH --job-name=executors_sbatch_declaration
#SBATCH --output=executors_sbatch_declaration.out
#SBATCH --error=executors_sbatch_declaration.err
##### END OF SCHEDULER DIRECTIVES #####
# Content of run section
hostname
***** End of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/executor_scheduler/
↳ executors_sbatch_declaration/17f960fc/executors_sbatch_declaration.sh_
↳ *****
----- executors_sbatch_declaration (ID: 17627bec-636c-42d2-a55b-
↳ 0429bb94a7d7) -----
executor: generic.local.sh
description: Declaring env and vars by executors section
state: PASS
returncode: 0
runtime: 0.027592
starttime: 2021/09/09 15:55:02
endtime: 2021/09/09 15:55:02
***** Start of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/executor_scheduler/
↳ executors_sbatch_declaration/17627bec/executors_sbatch_declaration.sh_
↳ *****
#!/bin/bash
##### START OF SCHEDULER DIRECTIVES #####
```

(continues on next page)

(continued from previous page)

```

#SBATCH -n 8
#SBATCH -N 1
#SBATCH -t 60
#SBATCH --job-name=executors_sbatch_declaration
#SBATCH --output=executors_sbatch_declaration.out
#SBATCH --error=executors_sbatch_declaration.err
##### END OF SCHEDULER DIRECTIVES #####
# Content of run section
hostname
***** End of Test Path: /home/docs/checkouts/readthedocs.org/user_
↳ builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/executor_scheduler/
↳ executors_sbatch_declaration/17627bec/executors_sbatch_declaration.sh_
↳ *****

```

Cray Burst Buffer and Data Warp

You can also define BB and DW directives in the `executors` field to override cray burst buffer and data warp settings per executor. buildtest will use the fields BB and DW and insert the `#BB` and `#DW` directives in the job script. For more details see *Cray Burst Buffer & Data Warp*.

```

version: "1.0"
buildspecs:
  create_burst_buffer_multiple_executors:
    type: script
    executor: "generic.local.(sh|bash)"
    sbatch: ["-N 1", "-t 10", "-C knl"]
    description: Create a burst buffer for multiple executors
    tags: [jobs]
    executors:
      generic.local.sh:
        BB:
          - create_persistent name=buffer1 capacity=10GB access_mode=striped type=scratch
        DW:
          - persistentdw name=buffer1
      generic.local.bash:
        BB:
          - create_persistent name=buffer2 capacity=10GB access_mode=striped type=scratch
        DW:
          - persistentdw name=buffer2
    run: hostname

```

Status and Metrics Field

The *status* and *metrics* field are supported in *executors* which can be defined within the named executor. In this next example, we will define *generic.local.bash* to match test based on returncode **0** or **2** and define metrics named *firstname* that is assigned the value from variable **FIRST**. The second test using *generic.local.sh* will match returncode of **1** and define a metrics named *lastname* that will store the value defined by variable **LAST**.

```
version: "1.0"
buildspecs:
  status_returncode_by_executors:
    type: script
    executor: "generic.local.(bash|sh)"
    description: define status and metrics per executor type.
    tags: [tutorials]
    vars:
      FIRST: Michael
      LAST: Jackson
    run: echo "my name is $FIRST $LAST"

  executors:
    generic.local.bash:
      status:
        returncode: [0, 2]
      metrics:
        firstname:
          vars: "FIRST"
    generic.local.sh:
      status:
        returncode: 1
      metrics:
        lastname:
          vars: "LAST"
```

Now let's run this test and we will see the test using **generic.local.sh** will fail because we have a returncode mismatch even though both tests got a 0 returncode as its actual value.

```
$ buildtest build -b tutorials/script/status_by_executors.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:03
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/script/status_by_executors.yml
```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪ -----+
| Discovered Buildsspecs                                     ↪
↪                                     |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ script/status_by_executors.yml |
+-----+
↪ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ script/status_by_executors.yml: VALID

Total builder objects created: 2
builders: [status_returncode_by_executors/8af6a3ff, status_returncode_by_executors/
↪ 912c478d]

name                id                description                                     ↪
↪ buildsspecs
-----
↪ -----
status_returncode_by_executors 8af6a3ff define status and metrics per executor type. /
↪ home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ script/status_by_executors.yml
status_returncode_by_executors 912c478d define status and metrics per executor type. /
↪ home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ script/status_by_executors.yml

+-----+
| Stage: Building Test |
+-----+

name                | id                | type | executor                | tags                                     ↪
↪ | testpath
-----+-----+-----+-----+-----
↪ +-----+-----+-----+-----+-----

```

(continues on next page)

(continued from previous page)

```

status_returncode_by_executors | 8af6a3ff | script | generic.local.bash | ['tutorials']_
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.bash/status_by_executors/status_returncode_by_executors/8af6a3ff/
↳ status_returncode_by_executors_build.sh
status_returncode_by_executors | 912c478d | script | generic.local.sh | ['tutorials']_
↳ | /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/
↳ tests/generic.local.sh/status_by_executors/status_returncode_by_executors/912c478d/
↳ status_returncode_by_executors_build.sh

+-----+
| Stage: Running Test |
+-----+

status_returncode_by_executors/912c478d: completed with returncode: 0
status_returncode_by_executors/912c478d: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳ status_by_executors/status_returncode_by_executors/912c478d/status_returncode_by_
↳ executors.out
status_returncode_by_executors/912c478d: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.sh/
↳ status_by_executors/status_returncode_by_executors/912c478d/status_returncode_by_
↳ executors.err
status_returncode_by_executors/912c478d: Checking returncode - 0 is matched in list [1]
status_returncode_by_executors/8af6a3ff: completed with returncode: 0
status_returncode_by_executors/8af6a3ff: Writing output file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ status_by_executors/status_returncode_by_executors/8af6a3ff/status_returncode_by_
↳ executors.out
status_returncode_by_executors/8af6a3ff: Writing error file: /home/docs/checkouts/
↳ readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/
↳ status_by_executors/status_returncode_by_executors/8af6a3ff/status_returncode_by_
↳ executors.err
status_returncode_by_executors/8af6a3ff: Checking returncode - 0 is matched in list [0,
↳ 2]

-----
Launching test: status_returncode_by_executors
Test ID: 8af6a3ff-a9d9-4db7-8782-6cc477deb384
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.bash/status_by_executors/status_returncode_by_executors/
↳ 8af6a3ff/status_returncode_by_executors_build.sh

-----
Launching test: status_returncode_by_executors
Test ID: 912c478d-85ba-4ef7-b1f4-48ca351a1cfb
Executor Name: generic.local.sh
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests/generic.local.sh/status_by_executors/status_returncode_by_executors/
↳ 912c478d/status_returncode_by_executors_build.sh

+-----+
| Stage: Test Summary |
+-----+

```

(continues on next page)

(continued from previous page)

name	id	executor	status	returncode_
↪match regex_match runtime_match returncode runtime				
-----+-----+-----+-----+-----				
↪ status_returncode_by_executors	8af6a3ff	generic.local.bash	PASS	True
↪ False False		0 0.020059		
status_returncode_by_executors	912c478d	generic.local.sh	FAIL	False
↪ False False		0 0.013939		

Passed Tests: 1/2 Percentage: 50.000%
Failed Tests: 1/2 Percentage: 50.000%

Writing Logfile to: /tmp/buildtest_78rzfc0s.log
A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log

Now let's see the test results by inspecting the metrics field using `buildtest report`. We see one test has the metrics name **firstname=Michael** and second test has **lastname=Jackson**.

```
$ buildtest report --format id,name,metrics --filter name=status_returncode_by_executors
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/var/report.json
```

id	name	metrics
8af6a3ff	status_returncode_by_executors	firstname=Michael
912c478d	status_returncode_by_executors	lastname=Jackson

run_only

The `run_only` property is used for running test given a specific condition has met. For example, you may want a test to run only if its particular system (Linux, Darwin), operating system, scheduler, etc...

run_only - user

buildtest will skip test if any of the conditions are not met. Let's take an example in this builds spec we define a test name **run_only_as_root** that requires **root** user to run test. The **run_only** is a property of key/value pairs and **user** is one of the field. buildtest will only build & run test if current user matches user field. We detect current user using `$USER` and match with input field `user`. buildtest will skip test if there is no match.

```
version: "1.0"
buildspecs:
  run_only_as_root:
```

(continues on next page)

(continued from previous page)

```

description: "This test will only run if current user is root"
executor: generic.local.bash
type: script
tags: ["tutorials"]
run_only:
  user: root
run: echo $USER

```

Now if we run this test we see buildtest will skip test **run_only_as_root** because current user is not root.

```

$ buildtest build -b tutorials/root_user.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:03
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b tutorials/root_user.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳ root_user.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

[run_only_as_root][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/tutorials/root_user.yml]: test is skipped because this test is expected to run
↳ as user: root but detected user: None.

```

(continues on next page)

(continued from previous page)

```
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳root_user.yml: VALID
Unable to create any builder objects
```

run_only - platform

Similarly, we can run test if it matches target platform. In this example we have two tests **run_only_platform_darwin** and **run_only_platform_linux** that are run if target platform is Darwin or Linux. This is configured using `platform` field which is a property of `run_only` object. buildtest will match target platform using `platform.system()` with field **platform**, if there is no match buildtest will skip test. In this test, we define a python shell using `shell: python` and `run platform.system()`. We expect the output of each test to have **Darwin** and **Linux** which we match with stdout using regular expression.

```
version: "1.0"
buildspecs:
  run_only_platform_darwin:
    description: "This test will only run if target platform is Darwin"
    executor: generic.local.python
    type: script
    tags: ["tutorials"]
    run_only:
      platform: Darwin
    shell: python
    run: |
      import platform
      print(platform.system())
    status:
      regex:
        stream: stdout
        exp: "^Darwin$"

  run_only_platform_linux:
    description: "This test will only run if target platform is Linux"
    executor: generic.local.python
    type: script
    tags: ["tutorials"]
    run_only:
      platform: Linux
    shell: python
    run: |
      import platform
      print(platform.system())
    status:
      regex:
        stream: stdout
        exp: "^Linux"
```

This test was ran on a MacOS (Darwin) so we expect test **run_only_platform_linux** to be skipped.

```
$ buildtest build -b tutorials/run_only_platform.yml

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/07/06 18:54:27
buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.9.6
python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
Configuration File: /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/
↳ settings/config.yml
Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
↳ tutorials/run_only_platform.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+=====+
| /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/run_only_platform.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1
[run_only_platform_linux][Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/
↳ run_only_platform.yml]: test is skipped because this test is expected to run on_
↳ platform: Linux but detected platform: Darwin.

+-----+
| Stage: Parsing Buildsspecs |
+-----+

  schemafile          | validstate | buildspec
+-----+-----+
↳ script-v1.0.schema.json | True      | /Users/siddiq90/Documents/GitHubDesktop/
↳ buildtest/tutorials/run_only_platform.yml

name                  description
+-----+-----+
run_only_platform_darwin This test will only run if target platform is Darwin

+-----+
| Stage: Building Test |
+-----+
```

(continues on next page)

(continued from previous page)

```

name          | id          | type   | executor          | tags          |
↳testpath
-----+-----+-----+-----+-----+
↳
↳
run_only_platform_darwin | 964e3016 | script | generic.local.python | ['tutorials'] | /
↳Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/run_
↳only_platform/run_only_platform_darwin/3/run_only_platform_darwin_build.sh

+-----+
| Stage: Running Test |
+-----+

name          | id          | executor          | status   | returncode
-----+-----+-----+-----+-----+
run_only_platform_darwin | 964e3016 | generic.local.python | PASS    | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /var/folders/1m/_jjv09h17k37mkktwnmbkmj0002t_q/T/buildtest__md43sa1.
↳log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
↳Documents/GitHubDesktop/buildtest/buildtest.log

```

run_only - scheduler

buildtest can run test if a particular scheduler is available. In this example, we introduce a new field `scheduler` that is part of `run_only` property. This field expects one of the following values: `[lsf, slurm, cobalt, pbs]` and buildtest will check if target system supports detects the scheduler. In this example we require **lsf** scheduler because this test runs **bmgroup** which is a LSF binary.

Note: buildtest assumes scheduler binaries are available in `$PATH`, if no scheduler is found buildtest sets this to an empty list

```

version: "1.0"
buildspecs:
  show_host_groups:
    type: script
    executor: generic.local.bash

```

(continues on next page)

(continued from previous page)

```
description: Show information about host groups using bmgrou
tags: lsf
run_only:
  scheduler: lsf
run: bmgrou
```

If we build this test on a target system without LSF notice that buildtest skips test **show_host_groups**.

```
$ buildtest build -b general_tests/sched/lsf/bmgroups.yml
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:04
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ bin/buildtest build -b general_tests/sched/lsf/bmgroups.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/sched/lsf/bmgroups.yml |
+-----+
↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

[show_host_groups][/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/general_tests/sched/lsf/bmgroups.yml]: test is skipped because ['run_only']
↳ 'scheduler'] got value: lsf but detected scheduler: [].
Valid Buildsspecs: 1
Invalid Buildsspecs: 0
```

(continues on next page)

(continued from previous page)

```
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/general_
↳ tests/sched/lsf/bmggroups.yml: VALID
Unable to create any builder objects
```

run_only - linux_distro

buildtest can run test if it matches a Linux distro, this is configured using `linux_distro` field that is a list of Linux distros that is returned via `distro.id()`. In this example, we run test only if host distro is darwin.

```
version: "1.0"
buildspecs:
  run_only_macos_distro:
    type: script
    executor: generic.local.bash
    description: "Run test only if distro is darwin."
    tags: [mac]
    run_only:
      linux_distro:
        - darwin
    run: uname
    status:
      regex:
        stream: stdout
        exp: "^Darwin$"

  run_only_linux_distro:
    type: script
    executor: generic.local.bash
    description: "Run test only if distro is CentOS."
    tags: [mac]
    run_only:
      linux_distro:
        - centos
    run: uname
```

This test will run successfully because this was ran on a Mac OS (darwin) system.

```
$ buildtest build -b tutorials/run_only_distro.yml

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/07/06 18:54:28
buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.9.6
python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
Configuration File: /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/
↳ settings/config.yml
```

(continues on next page)

(continued from previous page)

```

Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
↳ tutorials/run_only_distro.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+=====+
| /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/run_only_distro.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1
[run_only_linux_distro][Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/run_
↳ only_distro.yml]: test is skipped because this test is expected to run on linux_
↳ distro: ['centos'] but detected linux distro: darwin.

+-----+
| Stage: Parsing Buildsspecs |
+-----+

  schemafilename | validstate | buildspec
-----+-----+-----
↳ script-v1.0.schema.json | True | /Users/siddiq90/Documents/GitHubDesktop/
↳ buildtest/tutorials/run_only_distro.yml

name | description
-----+-----
run_only_macos_distro | Run test only if distro is darwin.

+-----+
| Stage: Building Test |
+-----+

  name | id | type | executor | tags | testpath
-----+-----+-----+-----+-----+-----
↳ run_only_macos_distro | 9d4d0d97 | script | generic.local.bash | ['mac'] | /Users/
↳ siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.bash/run_only_
↳ distro/run_only_macos_distro/0/run_only_macos_distro_build.sh

+-----+

```

(continues on next page)

(continued from previous page)

```
| Stage: Running Test |
+-----+

name          | id          | executor          | status  | returncode
+-----+-----+-----+-----+-----+
run_only_macos_distro | 9d4d0d97 | generic.local.bash | PASS    | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /var/folders/1m/_jjv09h17k37mkktwnmbkmj0002t_q/T/buildtest_6asbja74.
↪ log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
↪ Documents/GitHubDesktop/buildtest/buildtest.log
```

3.4.2 Global Schema

The global schema is validated with for all buildsspecs is the top-level schema when defining a buildspec file.

Please refer to [Global Schema Documentation](#) that provides a summary .

Schema Definition

Shown below is the start of the schema definition for **global.schema.json**

```
{
  "$id": "global.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "global schema",
  "description": "buildtest global schema is validated for all buildsspecs. The global_
↪ schema defines top-level structure of buildspec and defintions that are inherited for_
↪ sub-schemas",
  "type": "object",
  "required": ["version", "buildspecs"],
  "additionalProperties": false,
```

This schema requires that every buildspec should have version and buildspecs fields. The version key is required to lookup an a sub-schema using the type field. The buildspecs is the start of test declaration.

Example Buildspec

```
version: "1.0"
buildspecs:
  hello_world:
    executor: generic.local.bash
    type: script
    tags: tutorials
    description: "hello world example"
    run: echo "hello world!"
maintainers:
- "@shahzebsiddiqui"
```

The field `version` `buildspecs` and `maintainers` are validated with `global.schema.json` using `jsonschema.validate` method. The test section within `hello_world` is validated by sub-schema by looking up schema based on `type` field.

Every sub-schema requires `type` field in this case, `type: script` informs buildtest to validate with the *Script Schema*. All type schemas have a version, currently buildtest supports **1.0** version for all type schemas. The `version: "1.0"` is used to select the version of the sub-schema, in this example we validate with the schema `script-v1.0.schema.json`.

To understand how buildtest validates the buildspec see *parsing buildspecs*.

Maintainers

The `maintainers` is an optional field that can be used to specify a list of test maintainers for a given buildspec. The `maintainers` property is used by buildtest to report *buildspecs by maintainers* when querying buildspec cache. You can also *filter buildspecs* by maintainers during building via `buildtest build --filter maintainers=<NAME>` if one wants to filter tests

In this example, we have two maintainers `@johndoe` and `@bobsmith`. The `maintainers` is a list of strings but must be unique names, generally this can be your name or preferably a github or gitlab handle.

```
version: "1.0"
buildspecs:
  foo_bar:
    type: script
    executor: generic.local.sh
    tags: tutorials
    description: "prints variable $FOO"
    vars:
      FOO: BAR
    run: echo $FOO
maintainers:
- "@johndoe"
- "@bobsmith"
```

Test Names

The **buildspecs** property is a JSON object that defines one or more test. This is defined in JSON as follows:

```
"buildspecs": {
  "type": "object",
  "description": "This section is used to define one or more tests (buildspecs). Each
↳ test must be unique name",
  "propertyNames": {
    "pattern": "^[A-Za-z_.][A-Za-z0-9_.]*$",
    "maxLength": 32
  }
}
```

The test names take the following pattern "^[A-Za-z_.][A-Za-z0-9_.]*\$" and limited to 32 characters. In previous example, the test name is **hello_world**. You must have unique testname in your **buildspecs** section, otherwise you will have an invalid buildspec file. The description field is used to document the test and limited to 80 characters.

Note: We refer to the entire YAML content as **buildspec file**, this is not to be confused with the **buildspecs** field.

Buildspec Structure

Shown below is an overview of buildspec file. In this diagram we define one test within buildspecs property named systemd_default_target. This test is using the script schema defined by type: script. The executor property is a required property that determines how test is run. The executors are defined in buildtest configuration see [Configuring buildtest](#) for more details.

The run property is used for defining content of script, this can a shell-script (bash,csh) or python script.

version: "1.0"	Schema Version
buildspecs:	Declaration of tests
systemd_default_target:	Name of Test
executor: generic.local.bash	Name of Executor
type: script	Schema Type
tags: [system]	Tag Name
description: check if default target is multi-user.target	Description of Test
run:	Script
if ["multi-user.target" == `systemctl get-default`]; then	
echo "multi-user is the default target";	
exit 0	
fi	
echo "multi-user is not the default target";	
exit 1	

Please proceed to [Buildspec Overview](#) to learn more about buildspecs.

3.4.3 Compiler Schema

The compiler schema is used for compilation of programs, currently we support single source file compilation. In order to use the compiler schema you must set `type: compiler` in your sub-schema. See [compiler schema docs](#)

Compilation Examples

We assume the reader has basic understanding of *Global Schema* validation. Shown below is the schema header definition for `compiler-v1.0.schema.json`:

```
{
  "$id": "compiler-v1.0.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "compiler schema version 1.0",
  "description": "The compiler schema is of ``type: compiler`` in sub-schema which is
  ↪ used for compiling and running programs",
  "type": "object",
  "required": [
    "type",
    "source",
    "compilers",
    "executor"
  ],
}
```

The required fields for compiler schema are **type**, **compilers**, **source** and **executor**.

Shown below is a test name `hello_f` that compiles Fortran code with GNU compiler.

```
version: "1.0"
buildspecs:
  hello_f:
    type: compiler
    description: "Hello World Fortran Compilation"
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/hello.f90"
    compilers:
      name: ["^(builtin_gcc)$"]
      default:
        gcc:
          fflags: -Wall
```

The `source` property is used to specify input program for compilation, this can be a file relative to buildspec file or an absolute path. In this example the source file `src/hello.f90` is relative to buildspec file. The `compilers` section specifies compiler configuration, the `name` field is required property which is used to search compilers based on regular expression. In this example we use the **builtin_gcc** compiler as regular expression which is the system gcc compiler provided by buildtest. The `default` section specifies default compiler configuration applicable to a specific compiler group.

Shown below is an example build for the buildspec example

```
$ buildtest build -b tutorials/compilers/gnu_hello_fortran.yml
```

(continues on next page)

(continued from previous page)

```

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/07/06 18:54:28
buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.9.6
python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
Configuration File: /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/
↳ settings/config.yml
Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
↳ tutorials/compiler/gnu_hello_fortran.yml

```

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

```

```

+-----+
↳ -----+
| Discovered Buildsspecs |
↳ |
+=====+
| /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compiler/gnu_hello_
↳ fortran.yml |
+-----+

```

```

↳ -----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

```

```

+-----+
| Stage: Parsing Buildsspecs |
+-----+

```

```

  schemafile          | validstate | buildspect
+-----+-----+-----+
↳ compiler-v1.0.schema.json | True      | /Users/siddiq90/Documents/GitHubDesktop/
↳ buildtest/tutorials/compiler/gnu_hello_fortran.yml

```

```

name      description
-----
hello_f   Hello World Fortran Compilation

```

```

+-----+
| Stage: Building Test |
+-----+

```

(continues on next page)

(continued from previous page)

```

name      | id          | type      | executor          | tags          |
↪ compiler | testpath
-----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
hello_f | 5e3d8b5f | compiler | generic.local.bash | ['tutorials', 'compile'] | builtin_
↪ gcc | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.bash/
↪ gnu_hello_fortran/hello_f/1/hello_f_build.sh

+-----+
| Stage: Running Test |
+-----+

name      | id          | executor          | status | returncode
-----+-----+-----+-----+-----+
hello_f | 5e3d8b5f | generic.local.bash | FAIL   |          127

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 0/1 Percentage: 0.000%
Failed Tests: 1/1 Percentage: 100.000%

Writing Logfile to: /var/folders/1m/_jjv09h17k37mkktwnmbkmj0002t_q/T/buildtest_ycbz5z6n.
↪ log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
↪ Documents/GitHubDesktop/buildtest/buildtest.log

```

The generated test for test name **hello_f** is the following:

```

#!/bin/bash

# name of executable
_EXEC=hello.f90.exe
# Compilation Line
gfortran -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compilers/
↪ src/hello.f90

# Run executable
./$_EXEC

```

buildtest will use compiler wrappers specified in your settings to build the test, however these values can be overridden in buildspec file which will be discussed later.

The builtin_gcc compiler is defined below this can be retrieved by running `buildtest config compilers`. The `-y` will display compilers in YAML format.

```
$ buildtest config compilers -y
gcc:
  builtin_gcc:
    cc: /usr/bin/gcc
    cxx: /usr/bin/g++
    fc: /usr/bin/gfortran
```

buildtest will compile and run the code depending on the compiler flags. buildtest, will detect the file extension of source file (source property) to detect programming language and finally generate the appropriate C, C++, or Fortran compilation based on language detected. In this example, buildtest detects a **.f90** file extension and determines this is a Fortran program.

Shown below is the file extension table buildtest uses for determining the programming language.

Table 1: File Extension Language Mapping

Language	File Extension
C	.c
C++	.cc .cxx .cpp .c++
Fortran	.f90 .F90 .f95 .f .F .FOR .for .FTN .ftn

Compiler Selection

buildtest selects compiler based on name property which is a list of regular expression applied for available compilers defined in buildtest configuration. In example below we select all compilers with regular expression `^(builtin_gcc|gcc)` that is specified in line name: `["^(builtin_gcc|gcc)"]`

```
version: "1.0"
buildspecs:
  vecadd_gnu:
    type: compiler
    description: Vector Addition example with GNU compiler
    tags: [tutorials, compile]
    executor: generic.local.bash
    source: src/vecAdd.c
    compilers:
      name: ["^(builtin_gcc|gcc)"]
      default:
        gcc:
          cflags: -fopenacc
          ldflags: -lm
```

Currently, we have 3 compilers defined in buildtest settings, shown below is a listing of all compilers. We used `buildtest config compilers find` to *detect compilers*.

```
$ buildtest config compilers
builtin_gcc
gcc/9.3.0-n7p74fd
gcc/10.2.0-37fmsw7
```

Note: This example may vary on your machine depending on compilers available via `module` command.

We expect buidtest to select all three compilers based on our regular expression. In the following build, notice we have three tests for `vecadd_gnu` one for each compiler:

```
$ buidtest build -b tutorials/compilers/vecadd.yml

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/06/10 21:52:32
buidtest path: /Users/siddiq90/Documents/GitHubDesktop/buidtest/bin/buidtest
buidtest version: 0.9.5
python path: /Users/siddiq90/.local/share/virtualenvs/buidtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buidtest/var/tests
Configuration File: /Users/siddiq90/.buidtest/config.yml
Command: /Users/siddiq90/Documents/GitHubDesktop/buidtest/bin/buidtest build -b_
↳tutorials/compilers/vecadd.yml
```

```
+-----+
| Stage: Discovering Buildsspecs |
+-----+
```

```
+-----+
| Discovered Buildsspecs |
+=====+
| /Users/siddiq90/Documents/GitHubDesktop/buidtest/tutorials/compilers/vecadd.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1
```

```
+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

schemafile	validstate	buildspec
↳ compiler-v1.0.schema.json	True	/Users/siddiq90/Documents/GitHubDesktop/
↳ buildtest/tutorials/compilers/vecadd.yml		

name	description
vecadd_gnu	Vector Addition example with GNU compiler
vecadd_gnu	Vector Addition example with GNU compiler
vecadd_gnu	Vector Addition example with GNU compiler

```
+-----+
| Stage: Building Test |
+-----+
```

(continues on next page)

(continued from previous page)

```

name          | id          | type      | executor          | tags          |
↪ compiler    | testpath
-----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+
vecadd_gnu | 6f6b16e1 | compiler | generic.local.bash | ['tutorials', 'compile'] |
↪ builtin_gcc | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↪ generic.local.bash/vecadd/vecadd_gnu/2/vecadd_gnu_build.sh
vecadd_gnu | a76dd163 | compiler | generic.local.bash | ['tutorials', 'compile'] | gcc/
↪ 9.3.0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪ local.bash/vecadd/vecadd_gnu/3/vecadd_gnu_build.sh
vecadd_gnu | 82360702 | compiler | generic.local.bash | ['tutorials', 'compile'] | gcc/
↪ 10.2.0-37fmsw7 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪ local.bash/vecadd/vecadd_gnu/4/vecadd_gnu_build.sh

+-----+
| Stage: Running Test |
+-----+

name          | id          | executor          | status  | returncode
-----+-----+-----+-----+-----+
vecadd_gnu | 6f6b16e1 | generic.local.bash | PASS    | 0
vecadd_gnu | a76dd163 | generic.local.bash | PASS    | 0
vecadd_gnu | 82360702 | generic.local.bash | PASS    | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 3/3 Percentage: 100.000%
Failed Tests: 0/3 Percentage: 0.000%

Writing Logfile to: /Users/siddiq90/buildtest/buildtest_b0jwyoyv.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
↪ Documents/GitHubDesktop/buildtest/buildtest.log

```

buildtest will use compiler settings including module configuration from buildtest settings (`config.yml`). In example below we show the compiler definitions for the three gcc compilers. The `module` section is the declaration of modules to load, by default we disable purge (`purge: False`) which instructs buildtest to not insert `module purge`. The `load` is a list of modules to load via `module load`.

Shown below is the compiler configuration.

```

1 compilers:
2   find:
3     gcc: ^(gcc)
4   compiler:
5     gcc:
6       builtin_gcc:

```

(continues on next page)

(continued from previous page)

```

7      cc: gcc
8      fc: gfortran
9      cxx: g++
10     gcc/9.3.0-n7p74fd:
11         cc: gcc
12         cxx: g++
13         fc: gfortran
14     module:
15         load:
16         - gcc/9.3.0-n7p74fd
17         purge: false
18     gcc/10.2.0-37fmsw7:
19         cc: gcc
20         cxx: g++
21         fc: gfortran
22     module:
23         load:
24         - gcc/10.2.0-37fmsw7
25         purge: false

```

If we take a closer look at the generated test we see the *module load* command in the test script.

```

1  #!/bin/bash
2
3
4  # name of executable
5  _EXEC=vecAdd.c.exe
6  # Loading modules
7  module load gcc/10.2.0-37fmsw7
8  # Compilation Line
9  gcc -fopenacc -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/
  ↪compilers/src/vecAdd.c -lm
10
11
12 # Run executable
13 ./$_EXEC

```

```

1  #!/bin/bash
2
3
4  # name of executable
5  _EXEC=vecAdd.c.exe
6  # Loading modules
7  module load gcc/9.3.0-n7p74fd
8  # Compilation Line
9  gcc -fopenacc -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/
  ↪compilers/src/vecAdd.c -lm
10
11
12 # Run executable
13 ./$_EXEC

```

Excluding Compilers

The `exclude` property is part of `compilers` section which allows one to exclude compilers upon discovery by name field. The `exclude` property is a list of compiler names that will be removed from test generation which is done prior to build phase. buildtest will exclude any compilers specified in `exclude` if they were found based on regular expression in `name` field. In this example, we slightly modified previous example by excluding `gcc/10.2.0-37fmsw7` compiler. This is specified by `exclude`: `[gcc/10.2.0-37fmsw7]`.

```
version: "1.0"
buildspecs:
  vecadd_gnu_exclude:
    type: compiler
    description: Vector Addition example with GNU compilers but exclude gcc@10.2.0
    tags: [tutorials, compile]
    executor: generic.local.bash
    source: src/vecAdd.c
    compilers:
      name: ["^(gcc)"]
      exclude: [gcc/10.2.0-37fmsw7]
      default:
        gcc:
          cflags: -fopenacc
          ldflags: -lm
```

Notice when we build this test, buildtest will exclude `gcc/10.2.0-37fmsw7` compiler and test is not created during build phase.

```
1 $ buildtest build -b tutorials/compilers/compiler_exclude.yml
2
3
4 User: siddiq90
5 Hostname: DOE-7086392.local
6 Platform: Darwin
7 Current Time: 2021/06/10 21:56:11
8 buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
9 buildtest version: 0.9.5
10 python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
11 python version: 3.7.3
12 Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
13 Configuration File: /Users/siddiq90/.buildtest/config.yml
14 Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
↳ tutorials/compilers/compiler_exclude.yml
15
16 +-----+
17 | Stage: Discovering Buildsspecs |
18 +-----+
19
20 +-----+
21 ↳ ----+
22 | Discovered Buildsspecs
23 ↳ |
24 +-----+
25 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compilers/compiler_exclude.
26 ↳ yml |
```

(continues on next page)

(continued from previous page)

```

24 +-----+
25 |<-----+
26 Discovered Buildsspecs: 1
27 Excluded Buildsspecs: 0
28 Detected Buildsspecs after exclusion: 1
29 Excluding compiler: gcc/10.2.0-37fmsw7 from test generation
30
31 +-----+
32 | Stage: Parsing Buildsspecs |
33 +-----+
34
35 schemafile          | validstate  | buildspec
36 -----+-----+-----+
37 |<-----+
38 compiler-v1.0.schema.json | True          | /Users/siddiq90/Documents/GitHubDesktop/
39 |<buildtest/tutorials/compilers/compiler_exclude.yml
40
41 name                description
42 -----+-----+-----+
43 vecadd_gnu_exclude  Vector Addition example with GNU compilers but exclude gcc@10.2.0
44
45 +-----+
46 | Stage: Building Test |
47 +-----+
48
49
50 name                | id          | type      | executor          | tags
51 |< | compiler          | testpath
52 -----+-----+-----+-----+-----+
53 |<+-----+
54 |<-----+
55 vecadd_gnu_exclude | a7373d09 | compiler | generic.local.bash | ['tutorials', 'compile
56 |<'] | gcc/9.3.0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
57 |<generic.local.bash/compiler_exclude/vecadd_gnu_exclude/0/vecadd_gnu_exclude_build.sh
58
59 +-----+
60 | Stage: Running Test |
61 +-----+
62
63 name                | id          | executor          | status  | returncode
64 -----+-----+-----+-----+-----+
65 vecadd_gnu_exclude | a7373d09 | generic.local.bash | PASS    | 0
66
67 +-----+
68 | Stage: Test Summary |
69 +-----+
70
71 Passed Tests: 1/1 Percentage: 100.000%
72 Failed Tests: 0/1 Percentage: 0.000%

```

(continues on next page)

(continued from previous page)

```

68 Writing Logfile to: /Users/siddiq90/buildtest/buildtest_4szlay_j.log
69 A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
71 ↪Documents/GitHubDesktop/buildtest/buildtest.log

```

Compiler Defaults and Override Default Settings

Sometimes you may want to set default compiler flags (**cflags**, **fflags**, **cxxflags**), preprocessor (**cppflags**) or linker flags (**ldflags**) for compiler group (gcc, intel, pgc, etc...). This can be achieved using the default property that is part of **compilers** section.

The default field is organized into compiler groups, in example below we set default C compiler flags (cflags: -O1). In addition, we can override default settings using the config property where one must specify the compiler name to override. In example below we can override compiler settings for gcc/9.3.0-n7p74fd to use -O2 and gcc/10.2.0-37fmsw7 to use -O3 for **cflags**.

```

version: "1.0"
buildspecs:
  hello_c:
    type: compiler
    description: "Hello World C Compilation"
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/hello.c"
    compilers:
      name: ["^(builtin_gcc|gcc)"]
      default:
        gcc:
          cflags: -O1
      config:
        gcc/9.3.0-n7p74fd:
          cflags: -O2
        gcc/10.2.0-37fmsw7:
          cflags: -O3

```

Next we run this test, and we get three tests for test name **hello_c**.

```

$ buildtest build -b tutorials/compilers/gnu_hello_c.yml

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/06/10 22:00:08
buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.9.5
python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
Configuration File: /Users/siddiq90/.buildtest/config.yml
Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
↪tutorials/compilers/gnu_hello_c.yml

```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+=====+
| /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compilers/gnu_hello_c.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

  schemafile          | validstate | buildspec
-----+-----+-----
↪ compiler-v1.0.schema.json | True      | /Users/siddiq90/Documents/GitHubDesktop/
↪ buildtest/tutorials/compilers/gnu_hello_c.yml

name      description
-----
hello_c   Hello World C Compilation
hello_c   Hello World C Compilation
hello_c   Hello World C Compilation

+-----+
| Stage: Building Test |
+-----+

  name      | id          | type      | executor          | tags          |
↪ compiler  |             | testpath  |                   |               |
-----+-----+-----+-----+-----+
↪
↪
hello_c | afa92b9d | compiler | generic.local.bash | ['tutorials', 'compile'] | builtin_
↪ gcc    | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.
↪ bash/gnu_hello_c/hello_c/2/hello_c_build.sh
hello_c | 498010d3 | compiler | generic.local.bash | ['tutorials', 'compile'] | gcc/9.3.
↪ 0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.
↪ bash/gnu_hello_c/hello_c/3/hello_c_build.sh
hello_c | ee753488 | compiler | generic.local.bash | ['tutorials', 'compile'] | gcc/10.
↪ 2.0-37fmsw7 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪ local.bash/gnu_hello_c/hello_c/4/hello_c_build.sh

```

(continues on next page)

(continued from previous page)

```

+-----+
| Stage: Running Test |
+-----+

name      | id          | executor           | status  | returncode
+-----+-----+-----+-----+-----+
hello_c   | afa92b9d   | generic.local.bash | PASS    | 0
hello_c   | 498010d3   | generic.local.bash | PASS    | 0
hello_c   | ee753488   | generic.local.bash | PASS    | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 3/3 Percentage: 100.000%
Failed Tests: 0/3 Percentage: 0.000%

Writing Logfile to: /Users/siddiq90/buildtest/buildtest_dtyx0ags.log
A copy of logfile can be found at $BUILDTTEST_ROOT/buildtest.log - /Users/siddiq90/
↳ Documents/GitHubDesktop/buildtest/buildtest.log

Writing Logfile to: /private/tmp/buildtest/buildtest_hh9k7vm6.log

```

If we inspect the following test, we see the compiler flags are associated with the compiler. The test below is for *builtin_gcc* which use the default `-O1` compiler flag as shown below.

```

1  #!/bin/bash
2
3
4  # name of executable
5  _EXEC=hello.c.exe
6  # Compilation Line
7  gcc -O1 -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compilers/
↳ src/hello.c
8
9
10 # Run executable
11 ./$_EXEC

```

The test for `gcc/10.2.0-37fmsw7` and `gcc/9.3.0-n7p74fd` have cflags `-O3` and `-O2` set in their respective tests.

```

1  #!/bin/bash
2
3
4  # name of executable
5  _EXEC=hello.c.exe
6  # Loading modules
7  module load gcc/10.2.0-37fmsw7

```

(continues on next page)

(continued from previous page)

```

8 # Compilation Line
9 gcc -O3 -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compiler/
  ↪src/hello.c
10
11
12 # Run executable
13 ./$_EXEC

```

```

1 #!/bin/bash
2
3
4 # name of executable
5 _EXEC=hello.c.exe
6 # Loading modules
7 module load gcc/9.3.0-n7p74fd
8 # Compilation Line
9 gcc -O2 -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compiler/
  ↪src/hello.c
10
11
12 # Run executable
13 ./$_EXEC

```

Setting environment variables

Environment variables can be set using `env` property which is a list of key/value pair to assign environment variables. This property can be used in default section within a compiler group. In example below we have an OpenMP Hello World example in C where we define `OMP_NUM_THREADS` environment variable which controls number of OpenMP threads to use when running program. In this example we use 2 threads for all gcc compiler group

```

version: "1.0"
buildspecs:
  openmp_hello_c_example:
    type: compiler
    description: OpenMP Hello World C example
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/hello_omp.c"
    compilers:
      name: ["^(gcc)"]
      default:
        gcc:
          cflags: -fopenmp
          env:
            OMP_NUM_THREADS: 2

```

Shown below is one of the generated test and notice that buildtest will set environment variable `OMP_NUM_THREADS`.

```

1 #!/bin/bash
2

```

(continues on next page)

(continued from previous page)

```

3
4 # name of executable
5 _EXEC=hello_omp.c.exe
6 # Declare environment variables
7 export OMP_NUM_THREADS=2
8
9
10 # Loading modules
11 module load gcc/10.2.0-37fmsw7
12 # Compilation Line
13 gcc -fopenmp -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/
14 ↪compilers/src/hello_omp.c
15
16 # Run executable
17 ./$_EXEC

```

Similarly, one can define environment variables at the compiler level in `config` section. buildtest will override value defined in `default` section. In this example, we make slight modification to the test, so that `gcc/10.2.0-37fmsw7` will use 4 threads when running program. This will override the default value of 2.

```

version: "1.0"
buildspecs:
  override_environmentvars:
    type: compiler
    description: override default environment variables
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/hello_omp.c"
    compilers:
      name: ["^(gcc)"]
      default:
        gcc:
          cflags: -fopenmp
          env:
            OMP_NUM_THREADS: 2
      config:
        gcc/10.2.0-37fmsw7:
          env:
            OMP_NUM_THREADS: 4

```

Next we build this test as follows:

```

$ buildtest build -b tutorials/compilers/envvar_override.yml

User: siddiq90
Hostname: DOE-7086392.local
Platform: Darwin
Current Time: 2021/06/10 22:04:19
buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.9.5

```

(continues on next page)

(continued from previous page)

```
python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
python version: 3.7.3
Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
Configuration File: /Users/siddiq90/.buildtest/config.yml
Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
↳ tutorials/compiler/envvar_override.yml
```

```
+-----+
| Stage: Discovering Buildsspecs |
+-----+
```

```
+-----+
↳ ---+
| Discovered Buildsspecs |
↳ |
+=====+
| /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compiler/envvar_override.
↳ yml |
+-----+
```

```
↳ ---+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1
```

```
+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

```

  schemafile          | validstate | buildspec
-----+-----+-----
↳ compiler-v1.0.schema.json | True      | /Users/siddiq90/Documents/GitHubDesktop/
↳ buildtest/tutorials/compiler/envvar_override.yml
```

```

name                  description
-----
override_environmentvars  override default environment variables
override_environmentvars  override default environment variables
```

```
+-----+
| Stage: Building Test |
+-----+
```

```

name                  | id          | type          | executor          | tags
↳ | compiler          | testpath
-----+-----+-----+-----+-----
↳ -----+-----+-----+-----+-----
↳ -----+-----+-----+-----+-----
```

(continues on next page)

(continued from previous page)

```

override_environmentvars | 72619a4b | compiler | generic.local.bash | ['tutorials',
↪ 'compile'] | gcc/9.3.0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↪ var/tests/generic.local.bash/envvar_override/override_environmentvars/0/override_
↪ environmentvars_build.sh
override_environmentvars | 31098506 | compiler | generic.local.bash | ['tutorials',
↪ 'compile'] | gcc/10.2.0-37fmsw7 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↪ var/tests/generic.local.bash/envvar_override/override_environmentvars/1/override_
↪ environmentvars_build.sh

+-----+
| Stage: Running Test |
+-----+

name | id | executor | status | returncode
-----+-----+-----+-----+-----+
override_environmentvars | 72619a4b | generic.local.bash | PASS | 0
override_environmentvars | 31098506 | generic.local.bash | PASS | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 2/2 Percentage: 100.000%
Failed Tests: 0/2 Percentage: 0.000%

Writing Logfile to: /Users/siddiq90/buildtest/buildtest_p3wdnl1t.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
↪ Documents/GitHubDesktop/buildtest/buildtest.log

```

Now let's inspect the test by running `buildtest inspect name override_environmentvars` and we notice there are two test records for `override_environmentvars` using `gcc/9.3.0-n7p74fd` and `gcc/10.2.0-37fmsw7`.

```

1 $ buildtest inspect name override_environmentvars
2 Reading Report File: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/report.json
3
4 {
5   "override_environmentvars": [
6     {
7       "id": "72619a4b",
8       "full_id": "72619a4b-3ed2-489c-aebd-2e0cacbf2d6a",
9       "description": "override default environment variables",
10      "schemafilename": "compiler-v1.0.schema.json",
11      "executor": "generic.local.bash",
12      "compiler": "gcc/9.3.0-n7p74fd",
13      "hostname": "DOE-7086392.local",
14      "user": "siddiq90",
15      "testroot": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪ local.bash/envvar_override/override_environmentvars/0",
16      "testpath": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↪ local.bash/envvar_override/override_environmentvars/0/stage/override_environmentvars.sh
↪ ",

```

(continues on next page)

(continued from previous page)

```

17     "stagedir": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/0/stage",
18     "command": "sh override_environmentvars_build.sh",
19     "outfile": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/0/override_environmentvars.out",
20     "errfile": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/0/override_environmentvars.err",
21     "buildspec_content": "version: \"1.0\"\nbuildspecs:\n  override_environmentvars:\n
↳ type: compiler\n    description: override default environment variables\n
↳ executor: generic.local.bash\n    tags: [tutorials, compile]\n    source: \"src/hello_
↳ omp.c\"\n    compilers:\n      name: [\"^(gcc)\"]\n      default:\n        gcc:\n
↳ cflags: -fopenmp\n      env:\n        OMP_NUM_THREADS: 2\n      config:\n
↳ gcc/10.2.0-37fmsw7:\n      env:\n        OMP_NUM_THREADS: 4",
22     "test_content": "#!/bin/bash\n_n_EXEC=hello_omp.c.exe\nexport OMP_NUM_THREADS=2\n
↳ module load gcc/9.3.0-n7p74fd\ngcc -fopenmp -o $_EXEC /Users/siddiq90/Documents/
↳ GitHubDesktop/buildtest/tutorials/compilers/src/hello_omp.c\n./$_EXEC",
23     "logpath": "/Users/siddiq90/buildtest/buildtest_p3wdn11t.log",
24     "tags": "tutorials compile",
25     "starttime": "2021/06/10 22:04:19",
26     "endtime": "2021/06/10 22:04:20",
27     "runtime": 0.727095,
28     "state": "PASS",
29     "returncode": 0,
30     "output": "Hello World from thread = 0\nHello World from thread = 1\n",
31     "error": "The following have been reloaded with a version change:\n  1) gcc/10.2.0-
↳ 37fmsw7 => gcc/9.3.0-n7p74fd\n",
32     "job": null,
33     "build_script": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↳ generic.local.bash/envvar_override/override_environmentvars/0/override_environmentvars_
↳ build.sh"
34   },
35   {
36     "id": "31098506",
37     "full_id": "31098506-2bbf-4a50-8386-2fcd5bcd5ff5",
38     "description": "override default environment variables",
39     "schemafilename": "compiler-v1.0.schema.json",
40     "executor": "generic.local.bash",
41     "compiler": "gcc/10.2.0-37fmsw7",
42     "hostname": "DOE-7086392.local",
43     "user": "siddiq90",
44     "testroot": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/1",
45     "testpath": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/1/stage/override_environmentvars.sh
↳ ",
46     "stagedir": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/1/stage",
47     "command": "sh override_environmentvars_build.sh",
48     "outfile": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/1/override_environmentvars.out",
49     "errfile": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.
↳ local.bash/envvar_override/override_environmentvars/1/override_environmentvars.err",

```

(continues on next page)

(continued from previous page)

```

50     "buildspec_content": "version: \"1.0\"\\nbuildspecs:\\n  override_environmentvars:\\n
↪   type: compiler\\n    description: override default environment variables\\n
↪   executor: generic.local.bash\\n    tags: [tutorials, compile]\\n    source: \"src/hello_
↪   omp.c\"\\n    compilers:\\n      name: [\"^(gcc)\"]\\n      default:\\n        gcc:\\n
↪   cflags: -fopenmp\\n      env:\\n        OMP_NUM_THREADS: 2\\n      config:\\n
↪   gcc/10.2.0-37fmsw7:\\n      env:\\n        OMP_NUM_THREADS: 4\",
51     "test_content": "#!/bin/bash \\n_EXEC=hello_omp.c.exe\\nexport OMP_NUM_THREADS=4\\
↪   nmodule load gcc/10.2.0-37fmsw7\\ngcc -fopenmp -o $_EXEC /Users/siddiq90/Documents/
↪   GitHubDesktop/buildtest/tutorials/compilers/src/hello_omp.c\\n./$_EXEC\",
52     "logpath": "/Users/siddiq90/buildtest/buildtest_p3wdn11t.log",
53     "tags": "tutorials compile",
54     "starttime": "2021/06/10 22:04:20",
55     "endtime": "2021/06/10 22:04:20",
56     "runtime": 0.482645,
57     "state": "PASS",
58     "returncode": 0,
59     "output": "Hello World from thread = 1\\nHello World from thread = 3\\nHello World
↪   from thread = 2\\nHello World from thread = 0\\n\",
60     "error": "",
61     "job": null,
62     "build_script": "/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/
↪   generic.local.bash/envvar_override/override_environmentvars/1/override_environmentvars_
↪   build.sh"
63   }
64 ]
65 }

```

Tweak how test are passed

The `status` property can be used to determine how buildtest will pass the test. By default, buildtest will use `returncode` to determine if test PASS or FAIL with exitcode 0 as PASS and anything else is FAIL.

Sometimes, it may be useful check output of test to determine using regular expression. This can be done via `status` property. In this example, we define two tests, the first one defines `status` property in the default `gcc` group. This means all compilers that belong to gcc group will be matched with the regular expression.

In second example we override the `status regex` property for `gcc/10.2.0-37fmsw7`. We expect the test to produce an output of final result: `1.000000` so we expect one failure from `gcc/10.2.0-37fmsw7`.

```

version: "1.0"
buildspecs:
  default_status_regex:
    type: compiler
    description: Regular expression check in stdout for gcc group
    tags: [tutorials, compile]
    executor: generic.local.bash
    source: src/vecAdd.c
    compilers:
      name: [\"^(gcc)\"]
      default:
        gcc:
          cflags: -fopenacc

```

(continues on next page)

(continued from previous page)

```

ldflags: -lm
status:
  regex:
    stream: stdout
    exp: "^final result: 1.000000$"

override_status_regex:
  type: compiler
  description: Override regular expression for compiler gcc/10.2.0-37fmsw7
  tags: [tutorials, compile]
  executor: generic.local.bash
  source: src/vecAdd.c
  compilers:
    name: ["^(gcc)"]
    default:
      gcc:
        cflags: -fopenacc
        ldflags: -lm
        status:
          regex:
            stream: stdout
            exp: "^final result: 1.000000$"
    config:
      gcc/10.2.0-37fmsw7:
        status:
          regex:
            stream: stdout
            exp: "^final result: 0.99$"

```

If we build this test, notice that test id **9320ca41** failed which corresponds to gcc/10.2.0-37fmsw7 compiler test. The test fails because it fails to pass on regular expression even though we have a returncode of 0.

```

1  $ buildtest build -b tutorials/compilers/compiler_status_regex.yml
2
3
4  User: siddiq90
5  Hostname: DOE-7086392.local
6  Platform: Darwin
7  Current Time: 2021/06/10 22:08:03
8  buildtest path: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest
9  buildtest version: 0.9.5
10 python path: /Users/siddiq90/.local/share/virtualenvs/buildtest-KLOcDrW0/bin/python
11 python version: 3.7.3
12 Test Directory: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests
13 Configuration File: /Users/siddiq90/.buildtest/config.yml
14 Command: /Users/siddiq90/Documents/GitHubDesktop/buildtest/bin/buildtest build -b_
15 ↪tutorials/compilers/compiler_status_regex.yml
16
17 +-----+
18 | Stage: Discovering Buildsspecs |
19 +-----+

```

(continues on next page)

(continued from previous page)

```

20 +-----+
21 | Discovered Buildsspecs |
22 |
23 +-----+
24 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compiler_status_
25 | regex.yml |
26 +-----+
27 Discovered Buildsspecs: 1
28 Excluded Buildsspecs: 0
29 Detected Buildsspecs after exclusion: 1
30
31 +-----+
32 | Stage: Parsing Buildsspecs |
33 +-----+
34
35 schemafile          | validstate | buildspect
36 -----+-----+-----+
37 compiler-v1.0.schema.json | True      | /Users/siddiq90/Documents/GitHubDesktop/
38 buildtest/tutorials/compiler_status_regex.yml
39
40 name                | description
41 -----+-----+
42 default_status_regex | Regular expression check in stdout for gcc group
43 default_status_regex | Regular expression check in stdout for gcc group
44 override_status_regex | Override regular expression for compiler gcc/10.2.0-37fmsw7
45 override_status_regex | Override regular expression for compiler gcc/10.2.0-37fmsw7
46
47 +-----+
48 | Stage: Building Test |
49 +-----+
50
51
52 name                | id          | type      | executor          | tags
53 | compiler          | testpath
54 -----+-----+-----+-----+-----+
55
56 default_status_regex | a023a2c2 | compiler | generic.local.bash | ['tutorials',
57 'compile'] | gcc/9.3.0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
58 var/tests/generic.local.bash/compiler_status_regex/default_status_regex/0/default_
59 status_regex_build.sh
60 default_status_regex | 155865c3 | compiler | generic.local.bash | ['tutorials',
61 'compile'] | gcc/10.2.0-37fmsw7 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
62 var/tests/generic.local.bash/compiler_status_regex/default_status_regex/1/default_
63 status_regex_build.sh

```

(continues on next page)

(continued from previous page)

```

56  override_status_regex | 3411bddf | compiler | generic.local.bash | ['tutorials',
    ↳ 'compile'] | gcc/9.3.0-n7p74fd | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
    ↳ var/tests/generic.local.bash/compiler_status_regex/override_status_regex/0/override_
    ↳ status_regex_build.sh
57  override_status_regex | 295310a4 | compiler | generic.local.bash | ['tutorials',
    ↳ 'compile'] | gcc/10.2.0-37fmsw7 | /Users/siddiq90/Documents/GitHubDesktop/buildtest/
    ↳ var/tests/generic.local.bash/compiler_status_regex/override_status_regex/1/override_
    ↳ status_regex_build.sh
58
59  +-----+
60  | Stage: Running Test |
61  +-----+
62
63  name                | id          | executor          | status  | returncode
64  +-----+-----+-----+-----+-----+
65  default_status_regex | a023a2c2   | generic.local.bash | PASS    | 0
66  default_status_regex | 155865c3   | generic.local.bash | PASS    | 0
67  override_status_regex | 3411bddf   | generic.local.bash | PASS    | 0
68  override_status_regex | 295310a4   | generic.local.bash | FAIL    | 0
69
70  +-----+
71  | Stage: Test Summary |
72  +-----+
73
74  Passed Tests: 3/4 Percentage: 75.000%
75  Failed Tests: 1/4 Percentage: 25.000%
76
77
78  Writing Logfile to: /Users/siddiq90/buildtest/buildtest_hp7_gpbm.log
79  A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /Users/siddiq90/
    ↳ Documents/GitHubDesktop/buildtest/buildtest.log

```

Single Test Multiple Compilers

It's possible to run single test across multiple compilers (gcc, intel, cray, etc...). In the next example, we will build an OpenMP reduction test using gcc, intel and cray compilers. In this test, we use `name` field to select compilers that start with **gcc**, **intel** and **PrgEnv-cray** as compiler names. The `default` section is organized by compiler groups which inherits compiler flags for all compilers. OpenMP flag for gcc, intel and cray differ for instance one must use `-fopenmp` for gcc, `--qopenmp` for intel and `-h omp` for cray.

```

1  version: "1.0"
2  buildspecs:
3    reduction:
4      type: compiler
5      executor: local.bash
6      source: src/reduction.c
7      description: OpenMP reduction example using gcc, intel and cray compiler
8      tags: [openmp]
9      compilers:
10       name: ["^(gcc|intel|PrgEnv-cray)"]
11       default:

```

(continues on next page)

(continued from previous page)

```

12     all:
13         env:
14             OMP_NUM_THREADS: 4
15     gcc:
16         cflags: -fopenmp
17     intel:
18         cflags: -qopenmp
19     cray:
20         cflags: -h omp

```

In this example `OMP_NUM_THREADS` environment variable under the `all` section which will be used for all compiler groups. This example was built on Cori, we expect this test to run against every gcc, intel and PrgEnv-cray compiler module:

```

$ buildtest build -b buildsspecs/apps/openmp/reduction.yml

User: siddiq90
Hostname: cori02
Platform: Linux
Current Time: 2021/06/11 08:42:54
buildtest path: /global/homes/s/siddiq90/github/buildtest/bin/buildtest
buildtest version: 0.9.5
python path: /global/homes/s/siddiq90/.conda/envs/buildtest/bin/python
python version: 3.8.8
Test Directory: /global/u1/s/siddiq90/github/buildtest/var/tests
Configuration File: /global/u1/s/siddiq90/.buildtest/config.yml
Command: /global/homes/s/siddiq90/github/buildtest/bin/buildtest build -b buildsspecs/
↪ apps/openmp/reduction.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+=====+
| /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/apps/openmp/reduction.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

  schemafilename      | validstate | buildspec
+-----+-----+-----+
↪ compiler-v1.0.schema.json | True      | /global/u1/s/siddiq90/github/buildtest-cori/
↪ buildsspecs/apps/openmp/reduction.yml

```

(continues on next page)

(continued from previous page)

```

reduction | a6201c48 | compiler | cori.local.bash | ['openmp'] | gcc/8.3.0
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/29/reduction_build.sh
reduction | aa06b1be | compiler | cori.local.bash | ['openmp'] | gcc/9.3.0
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/30/reduction_build.sh
reduction | 02b8e7aa | compiler | cori.local.bash | ['openmp'] | gcc/10.1.0
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/31/reduction_build.sh
reduction | bd9abd7e | compiler | cori.local.bash | ['openmp'] | gcc/6.3.0
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/32/reduction_build.sh
reduction | 9409a86f | compiler | cori.local.bash | ['openmp'] | gcc/8.1.1-openacc-gcc-
→8-branch-20190215 | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/33/reduction_build.sh
reduction | b9700a0f | compiler | cori.local.bash | ['openmp'] | PrgEnv-cray/6.0.5
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/34/reduction_build.sh
reduction | a605c970 | compiler | cori.local.bash | ['openmp'] | PrgEnv-cray/6.0.7
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/35/reduction_build.sh
reduction | 9ef915a9 | compiler | cori.local.bash | ['openmp'] | PrgEnv-cray/6.0.9
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/36/reduction_build.sh
reduction | 4f9e4242 | compiler | cori.local.bash | ['openmp'] | intel/19.0.3.199
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/37/reduction_build.sh
reduction | e37befed | compiler | cori.local.bash | ['openmp'] | intel/19.1.2.254
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/38/reduction_build.sh
reduction | 1e9b0ab5 | compiler | cori.local.bash | ['openmp'] | intel/16.0.3.210
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/39/reduction_build.sh
reduction | 4e6d6f8a | compiler | cori.local.bash | ['openmp'] | intel/17.0.1.132
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/40/reduction_build.sh
reduction | ad1e44af | compiler | cori.local.bash | ['openmp'] | intel/17.0.2.174
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/41/reduction_build.sh
reduction | 49acf44b | compiler | cori.local.bash | ['openmp'] | intel/18.0.1.163
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/42/reduction_build.sh
reduction | 4192750c | compiler | cori.local.bash | ['openmp'] | intel/18.0.3.222
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/43/reduction_build.sh
reduction | 06584529 | compiler | cori.local.bash | ['openmp'] | intel/19.0.0.117
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/44/reduction_build.sh
reduction | 82fd9bab | compiler | cori.local.bash | ['openmp'] | intel/19.0.8.324
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/45/reduction_build.sh
reduction | 6140e8b4 | compiler | cori.local.bash | ['openmp'] | intel/19.1.0.166
→ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
→reduction/reduction/46/reduction_build.sh

```

(continues on next page)

(continued from previous page)

```

reduction | ac509e2e | compiler | cori.local.bash | ['openmp'] | intel/19.1.1.217
↳ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
↳reduction/reduction/47/reduction_build.sh
reduction | 9c39818e | compiler | cori.local.bash | ['openmp'] | intel/19.1.2.275
↳ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
↳reduction/reduction/48/reduction_build.sh
reduction | 2cb3acd1 | compiler | cori.local.bash | ['openmp'] | intel/19.1.3.304
↳ | /global/u1/s/siddiq90/github/buildtest/var/tests/cori.local.bash/
↳reduction/reduction/49/reduction_build.sh

+-----+
| Stage: Running Test |
+-----+

name      | id      | executor      | status  | returncode
+-----+
reduction | fd93fdcb | cori.local.bash | PASS    | 0
reduction | 43737191 | cori.local.bash | PASS    | 0
reduction | 6e2e95cd | cori.local.bash | PASS    | 0
reduction | c48a8d8d | cori.local.bash | PASS    | 0
reduction | a6201c48 | cori.local.bash | PASS    | 0
reduction | aa06b1be | cori.local.bash | PASS    | 0
reduction | 02b8e7aa | cori.local.bash | PASS    | 0
reduction | bd9abd7e | cori.local.bash | PASS    | 0
reduction | 9409a86f | cori.local.bash | PASS    | 0
reduction | b9700a0f | cori.local.bash | PASS    | 0
reduction | a605c970 | cori.local.bash | PASS    | 0
reduction | 9ef915a9 | cori.local.bash | PASS    | 0
reduction | 4f9e4242 | cori.local.bash | PASS    | 0
reduction | e37befed | cori.local.bash | PASS    | 0
reduction | 1e9b0ab5 | cori.local.bash | PASS    | 0
reduction | 4e6d6f8a | cori.local.bash | PASS    | 0
reduction | ad1e44af | cori.local.bash | PASS    | 0
reduction | 49acf44b | cori.local.bash | PASS    | 0
reduction | 4192750c | cori.local.bash | PASS    | 0
reduction | 06584529 | cori.local.bash | PASS    | 0
reduction | 82fd9bab | cori.local.bash | PASS    | 0
reduction | 6140e8b4 | cori.local.bash | PASS    | 0
reduction | ac509e2e | cori.local.bash | PASS    | 0
reduction | 9c39818e | cori.local.bash | PASS    | 0
reduction | 2cb3acd1 | cori.local.bash | PASS    | 0

+-----+
| Stage: Test Summary |
+-----+

Passed Tests: 25/25 Percentage: 100.000%
Failed Tests: 0/25 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_sq87154s.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /global/homes/s/
↳siddiq90/github/buildtest/buildtest.log

```

(continues on next page)

(continued from previous page)

If we inspect one of these tests from each compiler group (gcc, intel) we will see OMP_NUM_THREADS is set in all tests along with the appropriate compiler flag.

```

1  #!/bin/bash
2  _EXEC=reduction.c.exe
3  export OMP_NUM_THREADS=4
4  module load intel/19.1.3.304
5  icc -qopenmp -o $_EXEC /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/apps/
   ↪ openmp/src/reduction.c
6  ./$_EXEC

```

```

1  #!/bin/bash
2  _EXEC=reduction.c.exe
3  export OMP_NUM_THREADS=4
4  module load gcc/6.1.0
5  gcc -fopenmp -o $_EXEC /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/apps/
   ↪ openmp/src/reduction.c
6  ./$_EXEC

```

Customize Run Line

buildtest will define variable `_EXEC` in the job script that can be used to reference the generated binary. By default, buildtest will run the program standalone, but sometimes you may want to customize how job is run. This may include passing arguments or running binary through a job/mpi launcher. The `run` property expects user to specify how to launch program. buildtest will change directory to the called script before running executable. The compiled executable will be present in local directory which can be accessed via `./$_EXEC`. In example below we pass arguments `1 3 5` for gcc group and `100 200` for compiler `gcc/10.2.0-37fmsw7`.

```

version: "1.0"
buildspecs:
  custom_run_by_compilers:
    type: compiler
    description: Customize binary launch based on compiler
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/argc.c"
    compilers:
      name: ["^(builtin_gcc|gcc)"]
      default:
        gcc:
          run: ./$_EXEC 1 3 5
        config:
          gcc/10.2.0-37fmsw7:
            run: ./$_EXEC 100 120

```

If we build this test and see generated test, we notice buildtest customized the run line for launching binary. buildtest will directly replace content in `run` section into the shell-script. If no `run` field is specified buildtest will run the binary in standalone mode (`./$_EXEC`).

```
1 #!/bin/bash
2
3
4 # name of executable
5 _EXEC=argc.c.exe
6 # Loading modules
7 module load gcc/10.2.0-37fmsw7
8 # Compilation Line
9 gcc -o $_EXEC /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/compilers/src/
  ↪ argc.c
10
11
12 # Run executable
13 ./$_EXEC 100 120
```

MPI Example

In this example we run a MPI Laplace code using 4 process on a KNL node using the intel/19.1.2.254 compiler. This test is run on Cori through batch queue system. We can define **#SBATCH** parameters using sbatch property. This program is compiled using mpiicc wrapper this can be defined using cc parameter.

Currently, buildtest cannot detect if program is serial or MPI to infer appropriate compiler wrapper. If cc wasn't specified, buildtest would infer *icc* as compiler wrapper for C program. This program is run using *srun* job launcher, we can control how test is executed using the run property. This test required we swap intel modules and load *impi/2020* module.

```
1 version: "1.0"
2 buildspecs:
3   laplace_mpi:
4     type: compiler
5     description: Laplace MPI code in C
6     executor: slurm.knl_debug
7     tags: ["mpi"]
8     source: src/laplace_mpi.c
9     compilers:
10       name: ["^(intel/19.1.2.254)$"]
11       default:
12         all:
13           sbatch: ["-N 1", "-n 4"]
14           run: srun -n 4 $_EXEC
15         intel:
16           cc: mpiicc
17           cflags: -O3
18       config:
19         intel/19.1.2.254:
20           module:
21             load: [impi/2020]
22             swap: [intel, intel/19.1.2.254]
```

Shown below is a sample build for this buildspec, buildtest will dispatch and poll job until its complete.

```
$ buildtest build -b buildsspecs/apps/mpi/laplace_mpi.yml
```

User: siddiq90
 Hostname: cori02
 Platform: Linux
 Current Time: 2021/06/11 09:11:16
 buildtest path: /global/homes/s/siddiq90/github/buildtest/bin/buildtest
 buildtest version: 0.9.5
 python path: /global/homes/s/siddiq90/.conda/envs/buildtest/bin/python
 python version: 3.8.8
 Test Directory: /global/u1/s/siddiq90/github/buildtest/var/tests
 Configuration File: /global/u1/s/siddiq90/.buildtest/config.yml
 Command: /global/homes/s/siddiq90/github/buildtest/bin/buildtest build -b buildsspecs/
 ↳ apps/mpi/laplace_mpi.yml

```
+-----+
| Stage: Discovering Buildsspecs |
+-----+
```

```
+-----+
| Discovered Buildsspecs |
+-----+
| /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/apps/mpi/laplace_mpi.yml |
+-----+
```

Discovered Buildsspecs: 1
 Excluded Buildsspecs: 0
 Detected Buildsspecs after exclusion: 1

```
+-----+
| Stage: Parsing Buildsspecs |
+-----+
```

schemafile	validstate	buildspec
↳ compiler-v1.0.schema.json	True	/global/u1/s/siddiq90/github/buildtest-cori/ ↳ buildsspecs/apps/mpi/laplace_mpi.yml

```
name      description
-----
```

laplace_mpi	Laplace MPI code in C
-------------	-----------------------

```
+-----+
| Stage: Building Test |
+-----+
```

name	id	type	executor	tags	compiler	↳
↳ testpath						

(continues on next page)

(continued from previous page)

```

-----+-----+-----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+-----+-----+
↪ -----+-----+-----+-----+-----+-----+-----+
laplace_mpi | a6087b86 | compiler | cori.slurm.knl_debug | ['mpi'] | intel/19.1.2.254 | ↪
↪ /global/u1/s/siddiq90/github/buildtest/var/tests/cori.slurm.knl_debug/laplace_mpi/
↪ laplace_mpi/0/laplace_mpi_build.sh

+-----+
| Stage: Running Test |
+-----+

[laplace_mpi] JobID: 43308598 dispatched to scheduler
name      | id      | executor      | status  | returncode
-----+-----+-----+-----+-----+
laplace_mpi | a6087b86 | cori.slurm.knl_debug | N/A     | -1

Polling Jobs in 30 seconds

Job Queue: [43308598]

Pending Jobs

-----+-----+-----+-----+-----+
| name      | executor      | jobID  | jobstate |
+-----+-----+-----+-----+-----+
| laplace_mpi | cori.slurm.knl_debug | 43308598 | COMPLETED |
+-----+-----+-----+-----+-----+

Polling Jobs in 30 seconds

Job Queue: []

Completed Jobs

-----+-----+-----+-----+-----+
| name      | executor      | jobID  | jobstate |
+-----+-----+-----+-----+-----+
| laplace_mpi | cori.slurm.knl_debug | 43308598 | COMPLETED |
+-----+-----+-----+-----+-----+

+-----+
| Stage: Final Results after Polling all Jobs |
+-----+

```

(continues on next page)

(continued from previous page)

name	id	executor	status	returncode
laplace_mpi	a6087b86	cori.slurm.knl_debug	PASS	0
+-----+				
Stage: Test Summary				
+-----+				
Passed Tests: 1/1 Percentage: 100.000%				
Failed Tests: 0/1 Percentage: 0.000%				
Writing Logfile to: /tmp/buildtest_wgptyp8v.log				
A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /global/homes/s/ ↪siddiq90/github/buildtest/buildtest.log				

The generated test is as follows, note that buildtest will insert the #SBATCH directives at the top of script, and module load are done before module swap command.

```

1  #!/bin/bash
2  #SBATCH -N 1
3  #SBATCH -n 4
4  #SBATCH --job-name=laplace_mpi
5  #SBATCH --output=laplace_mpi.out
6  #SBATCH --error=laplace_mpi.err
7  _EXEC=laplace_mpi.c.exe
8  module load impi/2020
9  module swap intel intel/19.1.2.254
10 mpiicc -O3 -o $_EXEC /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/apps/mpi/src/
   ↪laplace_mpi.c
11 srun -n 4 $_EXEC

```

The master script that buildtest will invoke is the following, notice that our generated script (shown above) is invoked via *sbatch* with its options. The options *sbatch -q debug --clusters=cori -C knl,quad,cache* was inserted by our executor configuration. We add the *--parsable* option for Slurm jobs in order to get the JobID when this script is invoked so that buildtest can poll the job.

```

1  #!/bin/bash
2  source /global/u1/s/siddiq90/github/buildtest/var/executor/cori.slurm.knl_debug/before_
   ↪script.sh
3  sbatch --parsable -q debug --clusters=cori -C knl,quad,cache /global/u1/s/siddiq90/
   ↪github/buildtest/var/tests/cori.slurm.knl_debug/laplace_mpi/laplace_mpi/0/stage/
   ↪laplace_mpi.sh
4  returncode=$?
5  exit $returncode

```

Pre/Post sections for build and run section

The compiler schema comes with `pre_build`, `post_build`, `pre_run` and `post_run` fields where you can insert commands before and after build or run section. The **build** section is where we compile code, and **run** section is where compiled binary is executed.

Shown below is an example buildspec with pre/post section.

```
version: "1.0"
buildspecs:
  pre_post_build_run:
    type: compiler
    description: example using pre_build, post_build, pre_run, post_run example
    executor: generic.local.bash
    tags: [tutorials, compile]
    source: "src/hello.c"
    compilers:
      name: ["^(builtin_gcc)$"]
      default:
        gcc:
          cflags: -Wall
        all:
          pre_build: |
            echo "This is a pre-build section"
            gcc --version
          post_build: |
            echo "This is post-build section"
          pre_run: |
            echo "This is pre-run section"
            export FOO=BAR
          post_run: |
            echo "This is post-run section"
```

The format of the test structure is as follows.

```
#!/{shebang path} -- defaults to #!/bin/bash depends on executor name (local.bash, local.
↪ sh)
{job directives} -- sbatch or bsub field
{environment variables} -- env field
{variable declaration} -- vars field
{module commands} -- modules field

{pre build commands} -- pre_build field
{compile program} -- build field
{post build commands} -- post_build field

{pre run commands} -- pre_run field
{run executable} -- run field
{post run commands} -- post_run field
```

The generated test for this buildspec is the following:

```
#!/bin/bash
```

(continues on next page)

(continued from previous page)

```

# name of executable
_EXECUTABLE=hello.c.exe
### START OF PRE BUILD SECTION ###
echo "This is a pre-build section"
gcc --version

### END OF PRE BUILD SECTION ###

# Compilation Line
gcc -Wall -o $_EXECUTABLE /Users/siddiq90/Documents/GitHubDesktop/buildtest/tutorials/
↳compilers/src/hello.c

### START OF POST BUILD SECTION ###
echo "This is post-build section"

### END OF POST BUILD SECTION ###

### START OF PRE RUN SECTION ###
echo "This is pre-run section"
export FOO=BAR

### END OF PRE RUN SECTION ###

# Run executable
./$_EXECUTABLE

### START OF POST RUN SECTION ###
echo "This is post-run section"

### END OF POST RUN SECTION ###

```

3.4.4 Spack Schema

Note: This feature is in active development.

buildtest can generate tests for the `spack` package manager which can be used if you want to install or test packages as part of a repeatable process. You must set `type: spack` property in builds spec to use the spack schema for validating the builds spec test. Currently, we have `spack-v1.0.schema.json` JSON schema that defines the structure of how tests are to be written in builds spec. Shown below is the schema header. The **required** properties are `type`, `executor` and `spack`.

```

"$id": "spack-v1.0.schema.json",
"$schema": "http://json-schema.org/draft-07/schema#",

```

(continues on next page)

(continued from previous page)

```
"title": "spack schema version 1.0",
"description": "The spack schema is referenced using ``type: spack`` which is used for
↳generating tests using spack package manager",
"type": "object",
"required": [
  "type",
  "executor",
  "spack"
],
```

Install Specs

Let's start off with a simple example where we create a test that can `spack install zlib`. Shown below is a test named `install_zlib`. The `spack` keyword is a JSON object, in this test we define the root of spack using the `root` keyword which informs buildtest where spack is located. buildtest will automatically check the path and source the startup script. The `install` field is a JSON object that contains a `specs` property which is a list of strings types that are name of spack packages to install. Each item in the `specs` property will be added as a separate `spack install` command.

The schema is designed to mimic spack commands which will be clear with more examples.

```
version: "1.0"
buildspecs:
  install_zlib:
    type: spack
    executor: generic.local.sh
    description: "Install zlib"
    tags: [spack]
    spack:
      root: $HOME/spack
      install:
        specs: ['zlib']
```

If you build this test and inspect the generated script, buildtest will source spack startup script - `source $SPACK_ROOT/share/spack/setup-env.sh` based on the `root` property. In this example, we have spack cloned in `$HOME/spack` which is `/Users/siddiq90/spack` and buildtest will find the startup script which is in `share/spack/setup-env.sh`.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack install zlib
```

Spack Environment

buildtest can generate scripts to make use of `spack environments` which can be useful if you want to install or test specs in an isolated environment.

Currently, we can create spack environment (`spack env create`) via name, directory and manifest file (`spack.yaml`, `spack.lock`) and pass any options to `spack env create` command. Furthermore, we can activate existing spack environment via name or directory using `spack env activate` and pass options to the command. buildtest can remove spack environments automatically before creating spack environment or one can explicitly specify by name.

Activate Spack Environment

In this next example, we will activate an existing environment `m4` and add spec for `m4` and concretize the spack environment. The `env` is an object that mimics the `spack env` command. The `activate` field maps to `spack env activate` command. The `name` property is of type: `string` which is name of spack environment you want to activate. The `specs` property in `env` section maps to `spack add <specs` instead of `spack install`.

The property `concretize: true` will run `spack concretize` command that is only available as part of the `env` object since this command is only applicable in spack environments.

```
version: "1.0"
buildspecs:
  concretize_m4_in_spack_env:
    type: spack
    executor: generic.local.sh
    description: "Concretize m4 in a spack environment named m4"
    tags: [spack]
    spack:
      root: $HOME/spack
      env:
        specs:
          - 'm4'
        activate:
          name: m4
        concretize: true
```

If we build this test and inspect the generated test we see that spack will activate a spack environment `m4`, add specs in spack environment via `spack add m4` and concretize the environment. The `concretize` is a boolean type, if its `true` we will run `spack concretize -f`, if its `false` this command will not be in script.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env activate m4
spack add m4
spack concretize -f
```

If we inspect the output file we see that `m4` was concretized in the spack environment.

```
==> Package m4 was already added to m4
==> Concretized m4
[+] volmsbn m4@1.4.19%apple-clang@11.0.3+sigsegv arch=darwin-bigsur-skylake
[+] bc6kuc4 ^libsigsegv@2.13%apple-clang@11.0.3 arch=darwin-bigsur-skylake
```

Create a Spack Environment by name

In this next example, we will create a spack environment named `m4_zlib` that will install `m4` and `zlib` spec. The **create** field is a JSON object that maps to `spack env create` command which can pass some arguments in the form of key/value pairs. The `name` property in **create** section is used to create a spack environment by name.

The `compiler_find: true` is a boolean that determines if we need to find compilers in spack via `spack compiler find`. This can be useful if you need to find compilers so spack can install specs with a preferred compiler otherwise spack may have issues concretizing or install specs. buildtest will run **spack compiler find** after sourcing spack.

Note: The `compiler_find` option may not be useful if your compilers are already defined in one of your configuration scopes or `spack.yaml` that is part of your spack environment.

The `option` field can pass any command line arguments to `spack install` command and this field is available for other properties.

```
version: "1.0"
buildspecs:
  install_m4_zlib_in_spack_env:
    type: spack
    executor: generic.local.sh
    description: "Install m4 and zlib in a spack environment named m4_zlib"
    tags: [spack]
    spack:
      root: $HOME/spack
      compiler_find: true
      env:
        create:
          name: 'm4_zlib'
        specs:
          - 'm4'
          - 'zlib'
        activate:
          name: m4_zlib
      concretize: true
    install:
      option: '--keep-prefix'
```

If we build this test and see generated test we see that buildtest will create a spack environment `m4_zlib` and activate the environment, add **m4** and **zlib**, concretize the environment and install the specs.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack compiler find
spack env create m4_zlib
spack env activate m4_zlib
spack add m4
spack add zlib
spack concretize -f
spack install --keep-prefix
```

Now let's examine the output of this test, shown below is the summary of this test, as you can see we have successfully installed **m4** and **zlib** in a spack environment `m4_zlib`.

```

==> Found no new compilers
==> Compilers are defined in the following files:
    /Users/siddiq90/.spack/darwin/compilers.yaml
==> Updating view at /Users/siddiq90/spack/var/spack/environments/m4_zlib/.spack-env/view
==> Created environment 'm4_zlib' in /Users/siddiq90/spack/var/spack/environments/m4_zlib
==> You can activate this environment with:
==>   spack env activate m4_zlib
==> Adding m4 to environment m4_zlib
==> Adding zlib to environment m4_zlib
==> Concretized m4
[+] volmsbn  m4@1.4.19%apple-clang@11.0.3+sigsegev arch=darwin-bigsur-skylake
[+] bc6kuc4   ^libsigsgev@2.13%apple-clang@11.0.3 arch=darwin-bigsur-skylake
==> Concretized zlib
- 2hw3hzd  zlib@1.2.11%apple-clang@11.0.3+optimize+pic+shared arch=darwin-bigsur-
  ↪skylake
==> Updating view at /Users/siddiq90/spack/var/spack/environments/m4_zlib/.spack-env/view
==> Installing environment m4_zlib
==> Installing zlib-1.2.11-2hw3hzdfy7e2ndzojgqoq472m5flsloj
==> No binary for zlib-1.2.11-2hw3hzdfy7e2ndzojgqoq472m5flsloj found: installing from
  ↪source
==> Fetching https://mirror.spack.io/_source-cache/archive/c3/
  ↪c3e5e9fdd5004dcb542feda5ee4f0ff0744628baf8ed2dd5d66f8ca1197cb1a1.tar.gz
==> No patches needed for zlib
==> zlib: Executing phase: 'install'
==> zlib: Successfully installed zlib-1.2.11-2hw3hzdfy7e2ndzojgqoq472m5flsloj
    Fetch: 0.84s. Build: 6.98s. Total: 7.82s.
[+] /Users/siddiq90/spack/opt/spack/darwin-bigsur-skylake/apple-clang-11.0.3/zlib-1.2.11-
  ↪2hw3hzdfy7e2ndzojgqoq472m5flsloj
==> Updating view at /Users/siddiq90/spack/var/spack/environments/m4_zlib/.spack-env/view

```

Creating Spack Environment from Directory

We can create spack environment from a directory using the `dir` property that is available as part of `create` and `activate` field. In this next example we create a spack environment in our `$HOME` directory and concretize `m4` in the spack environment

```

version: "1.0"
buildspecs:
  spack_env_directory:
    type: spack
    executor: generic.local.sh
    description: "Concretize m4 in a spack environment named m4"
    tags: [spack]
    spack:
      root: $HOME/spack
      env:
        create:
          dir: $HOME/spack-envs/m4
        activate:
          dir: $HOME/spack-envs/m4
      specs:

```

(continues on next page)

(continued from previous page)

```
- 'm4'
concretize: true
```

When creating spack environment using directory, buildtest will automatically add the `-d` option which is required when creating spack environments. However, one can also pass this using the `option` field. Shown below is the generated script for the above test.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env create -d /Users/siddiq90/spack-envs/m4
spack env activate -d /Users/siddiq90/spack-envs/m4
spack add m4
spack concretize -f
```

buildtest will create environment first followed by activating the spack environment.

Create Spack Environment from Manifest File (`spack.yaml`, `spack.lock`)

Spack can create environments from `spack.yaml` or `spack.lock` which can be used if you have a spack configuration that works for your system and want to write a builds spec. While creating a spack environment, you can use the `manifest` property to specify path to your `spack.yaml` or `spack.lock`.

Note: buildtest will not enforce that manifest names be `spack.yaml` or `spack.lock` since spack allows one to create spack environment from arbitrary name so long as it is a valid spack configuration.

Shown below is an example builds spec that generates a test from a manifest file. The `manifest` property is of type: string and this is only available as part of `create` property.

```
version: "1.0"
buildspecs:
  spack_env_create_from_manifest:
    type: spack
    executor: generic.local.sh
    description: "Create spack environment from spack.yaml"
    tags: [spack]
    spack:
      root: $HOME/spack
      env:
        create:
          name: 'manifest_example'
          manifest: "$BUILDTEST_ROOT/tutorials/spack/example/spack.yaml"
        activate:
          name: 'manifest_example'
      concretize: true
```

If we build this test and inspect the generated script we see `spack env create` command will create an environment `manifest_example` using the manifest file that we provided.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env create manifest_example /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↳ tutorials/spack/example/spack.yaml
```

(continues on next page)

(continued from previous page)

```
spack env activate manifest_example
spack concretize -f
```

Removing Spack Environments

buildtest can remove spack environments which can be used if you are periodically running the same test where one is creating the same environment. buildtest can automatically remove spack environment using the property `remove_environment` which will remove the environment before creating it with same name. This field is part of the `create` field and only works if one is creating spack environments by name.

Alternately, buildtest provides the `rm` field which can be used for removing environment explicitly. In the `rm` field, the `name` is a required field which is the name of the spack environment to remove. The `name` field is of type: `string`. Shown below are two example tests where we remove spack environment using the **`remove_environment`** and **`rm`** field.

```
version: "1.0"
buildspecs:
  remove_environment_automatically:
    type: spack
    executor: generic.local.sh
    description: "remove spack environment automatically before creating a new_
↪environment"
    tags: [spack]
    spack:
      root: $HOME/spack
      env:
        create:
          remove_environment: true
          name: remove_environment
        activate:
          name: remove_environment
      specs:
        - 'bzip2'
      concretize: true

  remove_environment_explicit:
    type: spack
    executor: generic.local.sh
    description: "remove spack environment explicitly using the 'rm' property"
    tags: [spack]
    spack:
      root: $HOME/spack
      env:
        rm:
          name: dummy
        create:
          name: dummy
        activate:
          name: dummy
      specs:
        - 'bzip2'
      concretize: true
```

If we look at the generated test, we notice that spack will remove environments names: **remove_environment**, **dummy**.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env rm -y remove_environment
spack env create remove_environment
spack env activate remove_environment
spack add bzip2
spack concretize -f
```

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env rm -y dummy
spack env create dummy
spack env activate dummy
spack add bzip2
spack concretize -f
```

Pre and Post Commands

The spack schema supports ability to write arbitrary shell script content using the `pre_cmds` and `post_cmds` field that are of type: `string` and buildtest will insert the content into the test exactly as it is defined by these two fields.

In this next example, we will test an installation of *zlib* by cloning spack from upstream and use `pre_cmds` field to specify where we will clone spack. In this example, we will clone spack under **/tmp**. Since we don't have a valid root of spack since test hasn't been run, we can ignore check for spack paths by specifying `verify_spack: false` which informs buildtest to skip spack path check. Generally, buildtest will raise an exception if path specified by `root` is invalid and if `$SPACK_ROOT/share/spack/setup-env.sh` doesn't exist since this is the file that must be sourced.

The `pre_cmds` are shell commands that are run before sourcing spack, whereas the `post_cmds` are run at the very end of the script. In the `post_cmds`, we will `spack find` that will be run after `spack install`. We remove spack root (`$SPACK_ROOT`) so that this test can be rerun again.

```
version: "1.0"
buildspecs:
  run_pre_post_commands:
    type: spack
    executor: generic.local.sh
    description: "Install zlib"
    tags: [spack]
    pre_cmds: |
      cd /tmp
      git clone https://github.com/spack/spack
    spack:
      root: /tmp/spack
      verify_spack: false
      install:
        specs: ['zlib']
    post_cmds: |
      spack find
      rm -rf $SPACK_ROOT
```

If we build this test and inspect the generated script we see the following

```
#!/bin/bash

##### START OF PRE COMMANDS #####
cd /tmp
git clone https://github.com/spack/spack

##### END OF PRE COMMANDS #####

source /private/tmp/spack/share/spack/setup-env.sh
spack install zlib

##### START OF POST COMMANDS #####
spack find
rm -rf $SPACK_ROOT
##### END OF POST COMMANDS #####
```

If we inspect the output, we see that `zlib` is installed as shown in output from `spack find`

```
==> Installing zlib-1.2.11-2hw3hzdfy7e2ndzojgqoq472m5flsloj
==> No binary for zlib-1.2.11-2hw3hzdfy7e2ndzojgqoq472m5flsloj found: installing from
↳ source
==> Fetching https://mirror.spack.io/_source-cache/archive/c3/
↳ c3e5e9fdd5004dcb542feda5ee4f0ff0744628baf8ed2dd5d66f8ca1197cb1a1.tar.gz
==> No patches needed for zlib
==> zlib: Executing phase: 'install'
==> zlib: Successfully installed zlib-1.2.11-2hw3hzdfy7e2ndzojgqoq472m5flsloj
    Fetch: 0.50s. Build: 5.90s. Total: 6.40s.
[+] /private/tmp/spack/opt/spack/darwin-bigsur-skylake/apple-clang-11.0.3/zlib-1.2.11-
↳ 2hw3hzdfy7e2ndzojgqoq472m5flsloj
-- darwin-bigsur-skylake / apple-clang@11.0.3 -----
zlib@1.2.11
```

Specifying Scheduler Directives

The spack schema supports all of the *scheduler directives* such as `sbatch`, `bsub`, `pbs`, `cobalt`, and `batch` property in the buildspec.

The directives are applied at top of script. Shown below is a toy example that will define directives using `sbatch` and `batch` property. Note, this test won't submit job to scheduler since we are not using the a slurm executor.

```
version: "1.0"
buildspecs:
  spack_sbatch_example:
    type: spack
    executor: generic.local.sh
    description: "sbatch directives can be defined in spack schema"
    tags: [spack]
    sbatch: ["-N 1", "-n 8", "-t 30"]
    spack:
```

(continues on next page)

(continued from previous page)

```

root: $HOME/spack
env:
  specs:
    - 'm4'
  activate:
    name: m4
    concretize: true

```

buildtest will generate the shell script with the job directives and set the name, output and error files based on name of test. If we build this test, and inspect the generated test we see that **#SBATCH** directives are written based on the **sbatch** and **batch** field.

```

#!/bin/bash

##### START OF SCHEDULER DIRECTIVES #####
#SBATCH -N 1
#SBATCH -n 8
#SBATCH -t 30
#SBATCH --job-name=spack_sbatch_example
#SBATCH --output=spack_sbatch_example.out
#SBATCH --error=spack_sbatch_example.err
##### END OF SCHEDULER DIRECTIVES #####

source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env activate m4
spack add m4
spack concretize -f

```

You can define *multiple executors* in your builds spec with spack schema via **executors**. This can be useful if you need to specify different scheduler directives based on executor type since your executor will map to a queue.

Shown below is an example builds spec that will specify sbatch directives for **generic.local.sh** and **generic.local.bash**

```

version: "1.0"
buildspecs:
  spack_sbatch_multi_executors:
    type: spack
    executor: "generic.local.(sh|bash)"
    description: "sbatch directives can be defined in spack schema"
    tags: [spack]
    executors:
      generic.local.sh:
        sbatch: ["-N 1", "-t 30"]
      generic.local.bash:
        sbatch: ["-N 8", "-t 15"]
    pre_cmds: |
      cd /tmp
      git clone https://github.com/spack/spack
    spack:
      root: /tmp/spack
      env:

```

(continues on next page)

(continued from previous page)

```

specs:
  - 'm4'
create:
  name: m4
activate:
  name: m4
concretize: true
post_cmds: rm -rf $SPACK_ROOT

```

Configuring Spack Mirrors

We can add mirrors in the spack instance or spack environment using the mirror property which is available in the spack and env section. If the mirror property is part of the env section, the mirror will be added to spack environment. The mirror is an object that expects a Key/Value pair where the key is the name of mirror and value is location of the spack mirror.

In this next example, we will define a mirror name **e4s** that points to <https://cache.e4s.io> as the mirror location. Internally, this translates to `spack mirror add e4s https://cache.e4s.io` command.

```

version: "1.0"
buildspecs:
  add_mirror:
    type: spack
    executor: generic.local.sh
    description: Declare spack mirror
    tags: [spack]
    spack:
      root: $HOME/spack
      mirror:
        e4s: https://cache.e4s.io
    post_cmds: |
      spack mirror list

  add_mirror_in_spack_env:
    type: spack
    executor: generic.local.sh
    description: Declare spack mirror in spack environment
    tags: [spack]
    spack:
      root: $HOME/spack
      env:
        create:
          name: spack_mirror
        activate:
          name: spack_mirror
        mirror:
          e4s: https://cache.e4s.io
    post_cmds: |
      spack mirror list

```

If we look at the generated script for both tests, we see that mirror is added for both tests. Note that one can have mirrors

defined in their `spack.yaml` or one of the `configuration scopes` defined by `spack`.

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack mirror add e4s https://cache.e4s.io
```

```
##### START OF POST COMMANDS #####
spack mirror list
##### END OF POST COMMANDS #####
```

```
#!/bin/bash
source /Users/siddiq90/spack/share/spack/setup-env.sh
spack env create spack_mirror
spack env activate spack_mirror
spack mirror add e4s https://cache.e4s.io
```

```
##### START OF POST COMMANDS #####
spack mirror list
##### END OF POST COMMANDS #####
```

Spack Test

Note: `spack test` requires version 0.16.0 or higher in order to use this feature.

`buildtest` can run tests using `spack test run` that can be used for testing installed specs with tests provided by `spack`. In order to run tests, you need to declare the `test` section which is of `type: object` in JSON and `run` is a required property. The `run` section maps to `spack test run` that is responsible for running tests for a list of specs that are specified using the `specs` property.

Upon running the tests, we can retrieve results using `spack test results` which is configured using the `results` property. The `results` property expects one to specify the `specs` or `suite` or both in order to retrieve results.

The `suite` property is used to retrieve test results based on suite name, whereas `specs` property can be used to retrieve based on spec format. Both properties are a list of string types.

In example below we install `bzip2` and run the test using `spack test run bzip2`.

```
version: "1.0"
buildspecs:
  spack_test:
    type: spack
    executor: generic.local.sh
    description: "Install bzip2 and run spack test and report results"
    tags: [spack]
    pre_cmds: |
      cd /tmp
      git clone https://github.com/spack/spack
    spack:
      root: /tmp/spack
      verify_spack: false
```

(continues on next page)

(continued from previous page)

```

install:
  specs: ['bzip2']
test:
  run:
    specs: ['bzip2']
    results:
      suite: ['bzip2']

post_cmds: |
  spack find
  rm -rf $SPACK_ROOT

```

If we look at the generated test, buildtest will automatically set `--alias` option to define name of suite, otherwise spack will generate a random text for suite name which you won't know at time of writing test that is required by spack test results to fetch the results.

```

#!/bin/bash

##### START OF PRE COMMANDS #####
cd /tmp
git clone https://github.com/spack/spack spack

##### END OF PRE COMMANDS #####

source /private/tmp/spack-test-no-env/share/spack/setup-env.sh
spack install bzip2
spack test run --alias bzip2 bzip2
spack test results bzip2

##### START OF POST COMMANDS #####
spack find
rm -rf $SPACK_ROOT
##### END OF POST COMMANDS #####

```

Shown below is the example output of this test.

```

==> libiconv: Executing phase: 'configure'
==> libiconv: Executing phase: 'build'
==> libiconv: Executing phase: 'install'
==> libiconv: Successfully installed libiconv-1.16-xgemfyqy3gsdz3lk7wy3ejudfaksja4x
  Fetch: 1.54s. Build: 33.03s. Total: 34.57s.
[+] /private/tmp/spack/opt/spack/darwin-bigsur-skylake/apple-clang-11.0.3/libiconv-1.16-
  xgemfyqy3gsdz3lk7wy3ejudfaksja4x
==> Installing diffutils-3.7-3dfrh6li733xxcenwyjhwyta7xkh3udq
==> No binary for diffutils-3.7-3dfrh6li733xxcenwyjhwyta7xkh3udq found: installing from
  source
==> Fetching https://mirror.spack.io/_source-cache/archive/b3/
  b3a7a6221c3dc916085f0d205abf6b8e1ba443d4dd965118da364a1dc1cb3a26.tar.xz
==> No patches needed for diffutils

```

(continues on next page)

(continued from previous page)

```
==> diffutils: Executing phase: 'autoreconf'
==> diffutils: Executing phase: 'configure'
==> diffutils: Executing phase: 'build'
==> diffutils: Executing phase: 'install'
==> diffutils: Successfully installed diffutils-3.7-3dfrh6li733xxcenwyjhwyta7xkh3udq
Fetch: 1.32s. Build: 52.35s. Total: 53.67s.
[+] /private/tmp/spack/opt/spack/darwin-bigsur-skylake/apple-clang-11.0.3/diffutils-3.7-
↳ 3dfrh6li733xxcenwyjhwyta7xkh3udq
==> Installing bzip2-1.0.8-avjwvsoaivuflugopwk4ap7rffhejxzu
==> No binary for bzip2-1.0.8-avjwvsoaivuflugopwk4ap7rffhejxzu found: installing from
↳ source
==> Fetching https://mirror.spack.io/_source-cache/archive/ab/
↳ ab5a03176ee106d3f0fa90e381da478ddae405918153cca248e682cd0c4a2269.tar.gz
==> Ran patch() for bzip2
==> bzip2: Executing phase: 'install'
==> bzip2: Successfully installed bzip2-1.0.8-avjwvsoaivuflugopwk4ap7rffhejxzu
Fetch: 1.42s. Build: 1.84s. Total: 3.26s.
[+] /private/tmp/spack/opt/spack/darwin-bigsur-skylake/apple-clang-11.0.3/bzip2-1.0.8-
↳ avjwvsoaivuflugopwk4ap7rffhejxzu
==> Spack test bzip2
==> Testing package bzip2-1.0.8-avjwvso
==> Results for test suite 'bzip2':
==> bzip2-1.0.8-avjwvso PASSED
-- darwin-bigsur-skylake / apple-clang@11.0.3 -----
bzip2@1.0.8
diffutils@3.7
libiconv@1.16
```

We can search for test results using the spec format instead of suite name. In the `results` property we can use `specs` field instead of `suite` property to specify a list of spec names to run. In spack, you can retrieve the results using `spack test results -- <spec>`, note that double dash `--` is in front of spec name. We can pass options to `spack test results` using the `option` property which is available for `results` and `run` property. Currently, spack will write test results in `$HOME/.spack/tests` and we can use `spack test remove` to clear all test results. This can be done in `buildspec` using the `remove_tests` field which is a boolean. If this is set to **True** buildtest will run `spack test remove -y` to remove all test suites before running the tests.

```
version: "1.0"
buildspecs:
  spack_test_results_specs_format:
    type: spack
    executor: generic.local.sh
    description: "Run spack test results with spec format"
    tags: [spack]
    pre_cmds: |
      cd /tmp
      git clone https://github.com/spack/spack
  spack:
    root: /tmp/spack
    verify_spack: false
    install:
      specs: ['bzip2']
    test:
```

(continues on next page)

(continued from previous page)

```

remove_tests: true
run:
  specs: ['bzip2']
results:
  option: '-l'
  specs: ['bzip2']
post_cmds: |
  spack find
  rm -rf $SPACK_ROOT

```

In the generated test, we see that buildtest will remove all testsuites using `spack test remove -y` and query results based on spec format. The options are passed into `spack test results` based on the `option` field specified under the `results` section.

```

#!/bin/bash

##### START OF PRE COMMANDS #####
cd /tmp
git clone https://github.com/spack/spack

##### END OF PRE COMMANDS #####

source /private/tmp/spack/share/spack/setup-env.sh
spack install bzip2
spack test remove -y
spack test run --alias bzip2 bzip2
spack test results -l -- bzip2

##### START OF POST COMMANDS #####
spack find
rm -rf $SPACK_ROOT
##### END OF POST COMMANDS #####

```

3.5 Configuring buildtest

3.5.1 Overview

We assume you are familiar with general concepts presented in [getting started](#) and your next step is to configure buildtest to run at your site. This guide will present you the necessary steps to get you started.

When you clone buildtest, we provide a [default configuration](#) that can be used to run on your laptop or workstation that supports Linux or Mac. The buildtest configuration uses a JSON schemafile `settings.schema.json`. for validating your configuration. We have published the schema guide for settings schema which you can find [here](#).

Which configuration file does buildtest read?

buildtest will read configuration files in the following order:

- Command line `buildtest -c <config>.yaml build`
- User Configuration - `$HOME/.buildtest/config.yaml`
- Default Configuration - `$BUILDTEST_ROOT/buildtest/settings/config.yaml`

Default Configuration

Buildtest comes with a default configuration that can be found at `buildtest/settings/config.yaml` relative to root of repo. At the start of buildtest execution, buildtest will load the configuration file and validate the configuration with JSON schema `settings.schema.json`. If it's fails to validate, buildtest will raise an error.

We recommend you copy the default configuration as a template to configure buildtest for your site. To get started you should copy the file in `$HOME/.buildtest/config.yaml`. Please run the following command:

```
$ cp $BUILDTEST_ROOT/buildtest/settings/config.yaml $HOME/.buildtest/config.yaml
```

Shown below is the default configuration provided by buildtest.

```
$ cat $BUILDTEST_ROOT/buildtest/settings/config.yaml
system:
  generic:
    # specify list of hostnames where buildtest can run for given system record
    hostnames: [".*"]

    # system description
    description: Generic System
    # specify module system used at your site (environment-modules, lmod)
    moduletool: N/A
    # boolean to determine if buildsspecs provided in buildtest repo should be loaded in_
↪buildspec cache
    load_default_buildspecs: True

  executors:
    # define local executors for running jobs locally
    local:
      bash:
        description: submit jobs on local machine using bash shell
        shell: bash
      sh:
        description: submit jobs on local machine using sh shell
        shell: sh
      csh:
        description: submit jobs on local machine using csh shell
        shell: csh
      zsh:
        description: submit jobs on local machine using zsh shell
        shell: zsh
        disable: true
      python:
        description: submit jobs on local machine using python shell
```

(continues on next page)

(continued from previous page)

```

    shell: python

    # compiler block
    compilers:
        # regular expression to search for compilers based on module pattern. Used with
        ↪ 'buildtest config compilers find' to generate compiler instance
        # find:
        # gcc: "^gcc)"
        # intel: "^intel)"
        # cray: "^craype)"
        # pgi: "^pgi)"
        # cuda: "^cuda)"
        # clang: "^clang)"

        # declare compiler instance which can be site-specific. You can let 'buildtest.
        ↪ config compilers find' generate compiler section
        compiler:
            gcc:
                builtin_gcc:
                    cc: gcc
                    fc: gfortran
                    cxx: g++

        # location of log directory
        # logdir: /tmp/

        # specify location where buildtest will write tests
        # testdir: /tmp

        # specify one or more directory where buildtest should load buildsspecs
        # buildspect_roots: []

    cdash:
        url: https://my.cdash.org/
        project: buildtest
        site: generic
        buildname: tutorials

```

As you can see the layout of configuration starts with keyword `system` which is used to define one or more systems. Your HPC site may contain more than one cluster, so you should define your clusters with meaningful names as this will impact when you reference *executors* in buildsspecs. In this example, we define one cluster called `generic` which is a dummy cluster used for running tutorial examples. The **required** fields in the system scope are the following:

```

"required": ["executors", "moduletool", "load_default_buildspecs", "hostnames", "compilers
↪"]

```

The `hostnames` field is a list of nodes that belong to the cluster where buildtest should be run. Generally, these hosts should be your login nodes in your cluster. buildtest will process **hostnames** field across all system entry using `re.match` until a hostname is found, if none is found we report an error.

In this example we defined two systems *machine*, *machine2* with the following hostnames.

```
system:
  machine1:
    hostnames: ['loca$', '^1DOE']
  machine2:
    hostnames: ['BOB|JOHN']
```

In this example, none of the host entries match with hostname **DOE-7086392.local** so we get an error since buildtest needs to detect a system before proceeding.

```
buildtest.exceptions.BuildTestError: "Based on current system hostname: DOE-7086392.
↪local we cannot find a matching system ['machine1', 'machine2'] based on current_
↪hostnames: {'machine1': ['loca$', '^1DOE'], 'machine2': ['BOB|JOHN']}"
```

Let's assume you we have a system named **mycluster** that should run on nodes **login1**, **login2**, and **login3**. You can specify hostnames as follows.

```
system:
  mycluster:
    hostnames: ["login1", "login2", "login3"]
```

Alternately, you can use regular expression to condense this list

```
system:
  mycluster:
    hostnames: ["login[1-3]"]
```

Configuring Module Tool

You should configure the **moduletool** property to the module-system installed at your site. Valid options are the following:

```
# environment-modules
moduletool: environment-modules

# for lmod
moduletool: lmod

# specify N/A if you don't have modules
moduletool: N/A
```

buildspec roots

buildtest can discover buildspec using **buildspec_roots** keyword. This field is a list of directory paths to search for buildspecs. For example we clone the repo <https://github.com/buildtesters/buildtest-cori> at **\$HOME/buildtest-cori** and assign this to **buildspec_roots** as follows:

```
buildspec_roots:
  - $HOME/buildtest-cori
```

This field is used with the **buildtest buildspec find** command. If you rebuild your buildspec cache via **--rebuild** option, buildtest will search for all buildspecs in directories specified by **buildspec_roots** property. buildtest will recursively find all **.yaml** extension and validate each buildspec with appropriate schema.

Load Default Buildspecs

By default buildtest will add the `$BUILDTEST_ROOT/tutorials` and `$BUILDTEST_ROOT/general_tests` to search path when searching for buildsspecs with `buildtest buildspect find` command. This can be configured via `load_default_buildspecs` property which expects a boolean value.

By default we enable this property, however in practice you would want to disable this `load_default_buildspecs: False` if you only care about running your facility tests.

What is an executor?

An executor is responsible for running the test and capture output/error file and return code. An executor can be local executor which runs tests on local machine or batch executor that can be modelled as partition/queue. A batch executor is responsible for **dispatching** job, then **poll** job until its finish, and **gather** job metrics from scheduler.

Executor Declaration

The `executors` is a JSON *object*, that defines one or more executors. The executors are grouped by their type followed by executor name. In this example we define two local executors `bash`, `sh` and one slurm executor called `regular`:

```
system:
  generic:
    executors:
      local:
        bash:
          shell: bash
          description: bash shell
        sh:
          shell: sh
          description: sh shell
      slurm:
        regular:
          queue: regular
```

The **LocalExecutors** are defined in section *local* where each executor must be unique name and they are referenced in buildspect using `executor` field in the following format:

```
executor: <system>.<type>.<name>
```

For instance, if a buildspect wants to reference the LocalExecutor `bash` from the *generic* cluster, you would specify the following in the buildspect:

```
executor: generic.local.bash
```

In our example configuration, we defined a `bash` executor as follows:

```
executors:
  # define local executors for running jobs locally
  local:
    bash:
      description: submit jobs on local machine using bash shell
      shell: bash
```

The local executors requires the `shell` key which takes the pattern `"^(/bin/bash|/bin/sh|/bin/csh|/bin/tcsh|/bin/zsh|sh|bash|csh|tcsh|zsh|python).*$"`. Any buildspec that references this executor will submit job using bash shell.

You can pass options to shell which will get passed into each job submission. For instance if you want all bash scripts to run in login shell you can specify `bash --login`:

```
executors:
  local:
    login_bash:
      shell: bash --login
```

Then you can reference this executor as `executor: generic.local.login_bash` and your tests will be submitted via `bash --login /path/to/test.sh`.

Once you define your executors, you can *query the executors* via `buildtest config executors` command.

Disabling an executor

buildtest will run checks for every executor instance depending on the executor type, for instance local executors such as `bash`, `sh`, `csh` executor will be checked to see if shell is valid by checking the path. If shell doesn't exist, buildtest will raise an error. You can circumvent this issue by disabling the executor via `disable` property. A disabled executor won't serve any jobs which means any buildspec that reference the executor won't create a test.

In this next example the executor `zsh` is disabled which can be used if you don't have **zsh** on your system

```
executors:
  local:
    zsh:
      shell: zsh
      disable: true
```

before_script for executors

Often times, you may want to run a set of commands for a group of tests before running a test. We can do this using this using the `before_script` field which is defined in each executor that is of string type that expects bash commands.

This can be demonstrated with an executor name **local.e4s** responsible for building [E4S Testsuite](#)

```
local:
  e4s:
    description: "E4S testsuite locally"
    shell: bash
    before_script: |
      cd $SCRATCH
      git clone https://github.com/E4S-Project/testsuite.git
      cd testsuite
      source /global/common/software/spackcp/luke-wyatt-testing/spack/share/spack/setup-
      ↪env.sh
      source setup.sh
```

The `e4s` executor attempts to clone E4S Testsuite in `$SCRATCH` and activate a spack environment and run the initialize script `source setup.sh`. buildtest will write a `before_script.sh` for every executor. This can be found in `var/executors` directory as shown below

```
$ tree var/executors/
var/executors/
|-- local.bash
|   |-- before_script.sh
|-- local.e4s
|   |-- before_script.sh
|-- local.python
|   |-- before_script.sh
|-- local.sh
|   |-- before_script.sh

4 directories, 4 files
```

The `before_script` field is available for all executors and if its not specified the file will be empty. Every test will source these scripts for the appropriate executor.

Cori @ NERSC

Shown below is the configuration file used at Cori.

```
$ wget -q -O - https://raw.githubusercontent.com/buildtesters/buildtest-cori/devel/
↪config.yml 2>&1
system:
  gerty:
    hostnames:
      - gert01.nersc.gov
    load_default_buildspecs: false
    moduletool: environment-modules
    executors:
      local:
        bash:
          description: submit jobs on local machine using bash shell
          shell: bash
        sh:
          description: submit jobs on local machine using sh shell
          shell: sh
        csh:
          description: submit jobs on local machine using csh shell
          shell: csh
        python:
          description: submit jobs on local machine using python shell
          shell: python
    compilers:
      compiler:
        gcc:
          builtin_gcc:
            cc: /usr/bin/gcc
            cxx: /usr/bin/g++
            fc: /usr/bin/gfortran
      cdash:
        url: https://my.cdash.org
```

(continues on next page)

(continued from previous page)

```
project: buildtest-cori
site: gerty

perlmutter:
  hostnames:
  - login*
  load_default_buildspecs: false
  moduletool: lmod
  executors:
    defaults:
      pollinterval: 60
      launcher: sbatch
      max_pend_time: 90
    local:
      bash:
        description: submit jobs on local machine using bash shell
        shell: bash
      sh:
        description: submit jobs on local machine using sh shell
        shell: sh
      csh:
        description: submit jobs on local machine using csh shell
        shell: csh
      python:
        description: submit jobs on local machine using python shell
        shell: python
  compilers:
    find:
      gcc: ^(gcc)
    compiler:
      gcc:
        builtin_gcc:
          cc: /usr/bin/gcc
          cxx: /usr/bin/g++
          fc: /usr/bin/gfortran

cdash:
  url: https://my.cdash.org
  project: buildtest-cori
  site: perlmutter

cori:
  hostnames:
  - cori*
  load_default_buildspecs: false
  moduletool: environment-modules
  cdash:
    url: https://my.cdash.org
    project: buildtest-cori
    site: cori
  executors:
    defaults:
      pollinterval: 30
```

(continues on next page)

(continued from previous page)

```

launcher: sbatch
max_pend_time: 300
local:
  bash:
    description: submit jobs on local machine using bash shell
    shell: bash
  sh:
    description: submit jobs on local machine using sh shell
    shell: sh
  csh:
    description: submit jobs on local machine using csh shell
    shell: csh
  python:
    description: submit jobs on local machine using python shell
    shell: python
  e4s:
    description: E4S testsuite locally
    shell: bash
    before_script: |
      module load e4s/20.10
      cd $SCRATCH/testsuite
      source setup.sh

slurm:
  haswell_debug:
    qos: debug
    cluster: cori
    options:
      - -C haswell
    description: debug queue on Haswell partition
  haswell_shared:
    qos: shared
    cluster: cori
    options:
      - -C haswell
    description: shared queue on Haswell partition
  haswell_regular:
    qos: regular
    cluster: cori
    options:
      - -C haswell
    description: normal queue on Haswell partition
  haswell_premium:
    qos: premium
    cluster: cori
    options:
      - -C haswell
    description: premium queue on Haswell partition
  haswell_flex:
    qos: flex
    cluster: cori
    options:

```

(continues on next page)

(continued from previous page)

```
- -C haswell
description: flex queue on Haswell partition
knl_flex:
description: overrun queue on KNL partition
qos: overrun
cluster: cori
options:
- -C knl
bigmem:
description: bigmem jobs
cluster: escori
qos: bigmem
max_pend_time: 300
xfer:
description: xfer qos jobs
qos: xfer
cluster: escori
options:
- -C haswell
compile:
description: compile qos jobs
qos: compile
cluster: escori
options:
- -N 1
knl_debug:
qos: debug
cluster: cori
options:
- -C knl,quad,cache
description: debug queue on KNL partition
knl_regular:
qos: normal
cluster: cori
options:
- -C knl,quad,cache
description: normal queue on KNL partition
knl_premium:
qos: premium
cluster: cori
options:
- -C knl,quad,cache
description: premium queue on KNL partition
knl_low:
qos: low
cluster: cori
options:
- -C knl,quad,cache
description: low queue on KNL partition
knl_overrun:
description: overrun queue on KNL partition
qos: overrun
```

(continues on next page)

(continued from previous page)

```

cluster: cori
options:
- -C knl
- --time-min=01:00:00
gpu:
description: submit jobs to GPU partition
options:
- -C gpu
cluster: escori
max_pend_time: 300
e4s:
description: E4S runner
cluster: cori
max_pend_time: 20000
options:
- -q regular
- -C knl
- -t 10
- -n 4
before_script: |
module load e4s/20.10
cd $SCRATCH/testsuite
source setup.sh

compilers:
find:
gcc: ^(gcc|PrgEnv-gnu)
cray: ^(PrgEnv-cray)
intel: ^(intel|PrgEnv-intel)
cuda: ^(cuda/)
upcxx: ^(upcxx)
compiler:
gcc:
builtin_gcc:
cc: /usr/bin/gcc
cxx: /usr/bin/g++
fc: /usr/bin/gfortran
PrgEnv-gnu/6.0.5:
cc: gcc
cxx: g++
fc: gfortran
module:
load:
- PrgEnv-gnu/6.0.5
purge: false
PrgEnv-gnu/6.0.7:
cc: gcc
cxx: g++
fc: gfortran
module:
load:
- PrgEnv-gnu/6.0.7

```

(continues on next page)

(continued from previous page)

```
    purge: false
PrgEnv-gnu/6.0.9:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - PrgEnv-gnu/6.0.9
    purge: false
gcc/6.1.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/6.1.0
    purge: false
gcc/7.3.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/7.3.0
    purge: false
gcc/8.1.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/8.1.0
    purge: false
gcc/8.2.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/8.2.0
    purge: false
gcc/8.3.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/8.3.0
    purge: false
gcc/9.3.0:
  cc: gcc
  cxx: g++
```

(continues on next page)

(continued from previous page)

```

    fc: gfortran
    module:
      load:
        - gcc/9.3.0
      purge: false
gcc/10.1.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/10.1.0
    purge: false
gcc/6.3.0:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/6.3.0
    purge: false
gcc/8.1.1-openacc-gcc-8-branch-20190215:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/8.1.1-openacc-gcc-8-branch-20190215
    purge: false
cray:
  PrgEnv-cray/6.0.5:
    cc: cc
    cxx: CC
    fc: ftn
    module:
      load:
        - PrgEnv-cray/6.0.5
      purge: false
  PrgEnv-cray/6.0.7:
    cc: cc
    cxx: CC
    fc: ftn
    module:
      load:
        - PrgEnv-cray/6.0.7
      purge: false
  PrgEnv-cray/6.0.9:
    cc: cc
    cxx: CC
    fc: ftn
    module:
      load:

```

(continues on next page)

(continued from previous page)

```
        - PrgEnv-cray/6.0.9
        purge: false
intel:
  PrgEnv-intel/6.0.5:
    cc: icc
    cxx: icpc
    fc: ifort
    module:
      load:
        - PrgEnv-intel/6.0.5
        purge: false
  PrgEnv-intel/6.0.7:
    cc: icc
    cxx: icpc
    fc: ifort
    module:
      load:
        - PrgEnv-intel/6.0.7
        purge: false
  PrgEnv-intel/6.0.9:
    cc: icc
    cxx: icpc
    fc: ifort
    module:
      load:
        - PrgEnv-intel/6.0.9
        purge: false
intel/19.0.3.199:
  cc: icc
  cxx: icpc
  fc: ifort
  module:
    load:
      - intel/19.0.3.199
    purge: false
intel/19.1.2.254:
  cc: icc
  cxx: icpc
  fc: ifort
  module:
    load:
      - intel/19.1.2.254
    purge: false
intel/16.0.3.210:
  cc: icc
  cxx: icpc
  fc: ifort
  module:
    load:
      - intel/16.0.3.210
    purge: false
intel/17.0.1.132:
```

(continues on next page)

(continued from previous page)

```
cc: icc
cxx: icpc
fc: ifort
module:
  load:
    - intel/17.0.1.132
  purge: false
intel/17.0.2.174:
cc: icc
cxx: icpc
fc: ifort
module:
  load:
    - intel/17.0.2.174
  purge: false
intel/18.0.1.163:
cc: icc
cxx: icpc
fc: ifort
module:
  load:
    - intel/18.0.1.163
  purge: false
intel/18.0.3.222:
cc: icc
cxx: icpc
fc: ifort
module:
  load:
    - intel/18.0.3.222
  purge: false
intel/19.0.0.117:
cc: icc
cxx: icpc
fc: ifort
module:
  load:
    - intel/19.0.0.117
  purge: false
intel/19.0.8.324:
cc: icc
cxx: icpc
fc: ifort
module:
  load:
    - intel/19.0.8.324
  purge: false
intel/19.1.0.166:
cc: icc
cxx: icpc
fc: ifort
module:
```

(continues on next page)

(continued from previous page)

```
    load:
      - intel/19.1.0.166
    purge: false
intel/19.1.1.217:
  cc: icc
  cxx: icpc
  fc: ifort
  module:
    load:
      - intel/19.1.1.217
    purge: false
intel/19.1.2.275:
  cc: icc
  cxx: icpc
  fc: ifort
  module:
    load:
      - intel/19.1.2.275
    purge: false
intel/19.1.3.304:
  cc: icc
  cxx: icpc
  fc: ifort
  module:
    load:
      - intel/19.1.3.304
    purge: false
cuda:
  cuda/9.2.148:
    cc: nvcc
    cxx: nvcc
    fc: None
    module:
      load:
        - cuda/9.2.148
      purge: false
  cuda/10.0.130:
    cc: nvcc
    cxx: nvcc
    fc: None
    module:
      load:
        - cuda/10.0.130
      purge: false
  cuda/10.1.105:
    cc: nvcc
    cxx: nvcc
    fc: None
    module:
      load:
        - cuda/10.1.105
      purge: false
```

(continues on next page)

(continued from previous page)

```
cuda/10.1.168:
  cc: nvcc
  cxx: nvcc
  fc: None
  module:
    load:
      - cuda/10.1.168
    purge: false
cuda/10.1.243:
  cc: nvcc
  cxx: nvcc
  fc: None
  module:
    load:
      - cuda/10.1.243
    purge: false
cuda/10.2.89:
  cc: nvcc
  cxx: nvcc
  fc: None
  module:
    load:
      - cuda/10.2.89
    purge: false
cuda/11.0.2:
  cc: nvcc
  cxx: nvcc
  fc: None
  module:
    load:
      - cuda/11.0.2
    purge: false
cuda/shifter:
  cc: nvcc
  cxx: nvcc
  fc: None
  module:
    load:
      - cuda/shifter
    purge: false
upcxx:
  upcxx/2019.9.0:
    cc: upcxx
    cxx: upcxx
    fc: None
    module:
      load:
        - upcxx/2019.9.0
      purge: false
  upcxx/2020.3.0:
    cc: upcxx
    cxx: upcxx
```

(continues on next page)

(continued from previous page)

```
fc: None
module:
  load:
    - upcxx/2020.3.0
  purge: false
upcxx/2020.3.2:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx/2020.3.2
    purge: false
upcxx/2020.3.8-snapshot:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx/2020.3.8-snapshot
    purge: false
upcxx/2020.10.0:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx/2020.10.0
    purge: false
upcxx/2020.11.0:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx/2020.11.0
    purge: false
upcxx/bleeding-edge:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx/bleeding-edge
    purge: false
upcxx/nightly:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx/nightly
```

(continues on next page)

(continued from previous page)

```

    purge: false
upcxx-bupc-narrow/2019.9.0:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-bupc-narrow/2019.9.0
    purge: false
upcxx-bupc-narrow/2020.3.0:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-bupc-narrow/2020.3.0
    purge: false
upcxx-bupc-narrow/2020.3.2:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-bupc-narrow/2020.3.2
    purge: false
upcxx-bupc-narrow/2020.3.8-snapshot:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-bupc-narrow/2020.3.8-snapshot
    purge: false
upcxx-bupc-narrow/2020.11.0:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-bupc-narrow/2020.11.0
    purge: false
upcxx-bupc-narrow/bleeding-edge:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-bupc-narrow/bleeding-edge
    purge: false
upcxx-cs267/2020.10.0:
  cc: upcxx
  cxx: upcxx

```

(continues on next page)

(continued from previous page)

```
fc: None
module:
  load:
    - upcxx-cs267/2020.10.0
  purge: false
upcxx-extras/2020.3.0:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-extras/2020.3.0
    purge: false
upcxx-extras/2020.3.8:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-extras/2020.3.8
    purge: false
upcxx-extras/master:
  cc: upcxx
  cxx: upcxx
  fc: None
  module:
    load:
      - upcxx-extras/master
    purge: false
```

Specifying QoS (Slurm)

At Cori, jobs are submitted via qos instead of partition so we model a slurm executor named by qos. The qos field instructs which Slurm QOS to use when submitting job. For example we defined a slurm executor named **haswell_debug** which will submit jobs to **debug** qos on the haswell partition as follows:

```
executors:
  slurm:
    haswell_debug:
      qos: debug
      cluster: cori
      options:
        - -C haswell
```

The cluster field specifies which slurm cluster to use (i.e `sbatch --clusters=<string>`). In-order to use bigmem, xfer, or gpu qos at Cori, we need to specify **escori** cluster (i.e `sbatch --clusters=escori`).

buildtest will detect slurm configuration and check qos, partition, cluster match with buildtest configuration. In addition, buildtest supports multi-cluster job submission and monitoring from remote cluster. This means if you specify cluster field buildtest will poll jobs using `sacct` with the cluster name as follows: `sacct -M <cluster>`.

The options field is use to specify any additional options to launcher (`sbatch`) on command line. For instance, slurm. gpu executor, we use the options: `-C gpu` to submit to Cori GPU cluster which requires `sbatch -M escori -C`

gpu. Any additional **#SBATCH** options are defined in buildspec for more details see [batch scheduler support](#).

Specify Slurm Partitions

You can specify slurm partitions instead of qos if your slurm cluster requires jobs to be submitted by partitions. This can be done via `partition` property. In this next example we define an executor name *regular_hsw* which maps to slurm partition **regular_hsw**.

```
regular_hsw:
  partition: regular_hsw
  description: regular haswell queue
```

buildtest will check if slurm partition is in **up** state before adding executor. buildtest will be performing these checks when validating configuration file and this avoids creating tests that reference a partition that is in **down** state. Internally, we are running the following command for every defined defined partition

```
$ sinfo -p regular_hsw -h -O available
up
```

Default Executor Settings

We can define default configurations for all executors using the `defaults` property.

```
executors:
  defaults:
    pollinterval: 10
    launcher: sbatch
    max_pend_time: 90
    account: nstaff
```

The *launcher* field is applicable for batch executors in this case, `launcher: sbatch` inherits **sbatch** as the job launcher for all slurm executors.

The `account: nstaff` will instruct buildtest to charge all jobs to account `nstaff` from Slurm Executors. The `account` option can be set in `defaults` field to all executors or defined per executor instance which overrides the default value.

Poll Interval

The `pollinterval` field is used to poll jobs at set interval in seconds when job is active in queue. The poll interval can be configured on command line using `buildtest build --poll-interval` which overrides the configuration value.

Note: `pollinterval`, `launcher` and `max_pend_time` have no effect on local executors.

Max Pend Time

The `max_pend_time` is **maximum** time job can be pending within an executor, if it exceeds the limit buildtest will cancel the job.

The `max_pend_time` option can be overridden per executor level for example the section below overrides the default to 300 seconds:

```
bigmem:
  description: bigmem jobs
  cluster: escori
  qos: bigmem
  max_pend_time: 300
```

The `max_pend_time` is used to cancel job only if job is pending in queue, it has no impact if job is running. buildtest starts a timer at job submission and every poll interval (`pollinterval` field) checks if job has exceeded `max_pend_time` only if job is pending. If job pendtime exceeds `max_pend_time` limit, buildtest will cancel job the job using the appropriate scheduler command like (`scancel`, `bkill`, `qdel`). Buildtest will remove cancelled jobs from poll queue, in addition cancelled jobs won't be reported in test report.

For more details on `max_pend_time` click [here](#).

PBS Executors

Note: buildtest PBS support relies on job history set because buildtest needs to query job after completion using `qstat -x`. This can be configured using `qmgr` by setting `set server job_history_enable=True`. For more details see section **13.15.5.1 Enabling Job History** in [PBS 2020.1 Admin Guide](#)

buildtest supports [PBS](#) scheduler which can be defined in the `executors` section. Shown below is an example configuration using one `pbs` executor named `workq`. The property `queue: workq` defines the name of PBS queue that is available in your system.

```
1 system:
2   generic:
3     hostnames: ['.*']
4
5   moduletool: N/A
6   load_default_buildspecs: True
7   executors:
8     defaults:
9       pollinterval: 10
10      launcher: qsub
11      max_pend_time: 30
12   pbs:
13     workq:
14       queue: workq
15   compilers:
16     compiler:
17       gcc:
18         default:
19           cc: /usr/bin/gcc
20           cxx: /usr/bin/g++
21           fc: /usr/bin/gfortran
```

buildtest will detect the PBS queues in your system and determine if queues are active and enabled before submitting job to scheduler. buildtest will run `qstat -Q -f -F json` command to check for queue state which reports in JSON format and check if queue has the fields `enabled: "True"` or `started: "True"` set in the queue definition. If these values are not set, buildtest will raise an exception.

Shown below is an example with one queue **workq** that is enabled and started.

```

1 $ qstat -Q -f -F json
2 {
3   "timestamp":1615924938,
4   "pbs_version":"19.0.0",
5   "pbs_server":"pbs",
6   "Queue":{
7     "workq":{
8       "queue_type":"Execution",
9       "total_jobs":0,
10      "state_count":"Transit:0 Queued:0 Held:0 Waiting:0 Running:0 Exiting:0_
↪ Begun:0 ",
11      "resources_assigned":{
12        "mem":"0kb",
13        "ncpus":0,
14        "nodect":0
15      },
16      "hasnodes":"True",
17      "enabled":"True",
18      "started":"True"
19    }
20  }
21 }
```

Configuring test directory

The default location where tests are written is `$BUILDTEST_ROOT/var/tests` where `$BUILDTEST_ROOT` is the root of buildtest repo. You may specify `testdir` in your configuration to instruct where tests can be written. For instance, if you want to write tests in `/tmp` you can set the following:

```
testdir: /tmp
```

Alternately, one can specify test directory via `buildtest build --testdir <path>` which has highest precedence and overrides configuration and default value.

Configuring log path

You can configure where buildtest will write logs using `logdir` property. For example, in example below buildtest will write log files `$HOME/Documents/buildtest/var/logs`. buildtest will resolve variable expansion to get real path on filesystem.

```
# location of log directory
logdir: $HOME/Documents/buildtest/var/logs
```

`logdir` is not required field in configuration, if it's not specified then buildtest will write logs based on `tempfile` library which may vary based on platform (Linux, Mac).

The buildtest logs will start with **buildtest_** followed by random identifier with a **.log** extension.

buildtest will write the same log file in **\$BUILDTEST_ROOT/buildtest.log** which can be used to fetch last build log. This can be convenient if you don't remember the directory path to log file.

CDASH Configuration

buildtest can be configured to push test to **CDASH**. The default configuration file provides a CDASH configuration for buildtest project is the following.

```
cdash:
  url: https://my.cdash.org/
  project: buildtest
  site: generic
  buildname: tutorials
```

The cdash section can be summarized as follows:

- **url**: URL to CDASH server
- **project**: Project Name in CDASH server
- **site**: Site name that shows up in CDASH entry. This should be name of your system name
- **buildname**: Build Name that shows up in CDASH, this can be any name you want.

The cdash settings can be used with `buildtest cdash` command. For more details see *CDASH Integration (buildtest cdash)*.

3.5.2 Defining Compilers at your site

buildtest provides a mechanism to declare compilers in your configuration file, this is defined in `compilers` top-level section. The compilers should reflect compilers installed at your site. The compilers are used if you are writing a builds spec with *compiler schema* that needs to reference a particular compiler. The compilers are declared within scope of a system since we assume compilers will vary across different HPC clusters.

Compiler Declaration

Shown below is a declaration of `builtin_gcc` provided by default.

```
compilers:
  compiler:
    gcc:
      builtin_gcc:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran
```

The compiler declaration is defined in section `compiler` followed by name of compiler in this case `gcc`. In the `gcc` section one can define all gnu compilers, which includes the name of the compiler in this example we call `builtin_gcc` as system compiler that defines C, C++ and Fortran compilers using `cc`, `cxx` and `fc`.

One can retrieve all compilers using `buildtest config compilers`, there are few options for this command.


```
$ buildtest config compilers --help
usage: buildtest [options] [COMMANDS] config compilers [-h] [-j] [-y] ...

optional arguments:
  -h, --help  show this help message and exit
  -j, --json  List compiler details in JSON format
  -y, --yaml  List compiler details in YAML format

subcommands:
  Find new compilers and add them to detected compiler section

  find      Find compilers
```

buildtest can represent compiler output in JSON, YAML using the `--json` and `--yaml`. Shown below is an example output with these options:

```
$ buildtest config compilers --json
{
  "gcc": {
    "builtin_gcc": {
      "cc": "/usr/bin/gcc",
      "cxx": "/usr/bin/g++",
      "fc": "/usr/bin/gfortran"
    }
  }
}

$ buildtest config compilers --yaml
gcc:
  builtin_gcc:
    cc: /usr/bin/gcc
    cxx: /usr/bin/g++
    fc: /usr/bin/gfortran

$ buildtest config compilers
builtin_gcc
```

Detect Compilers (Experimental Feature)

buildtest can detect compilers based on modulefiles and generate compiler section that way you don't have to specify each compiler manually. This can be done via `buildtest config compilers find` command. Buildtest expects a key/value mapping when searching compiler names and regular expression using `re.match` for discovering compiler modules.

This can be demonstrated, by defining search pattern in the `find` section that expects a dictionary of key/value mapping between compiler names and their module names.

In example, below we define a pattern for gcc modules as `^(gcc)` which will find all modules that start with name `gcc`.

```
compilers:
  find:
    gcc: "^(gcc)"
```

(continues on next page)

(continued from previous page)

```

compiler:
  gcc:
    builtin:
      cc: /usr/bin/gcc
      cxx: /usr/bin/g++
      fc: /usr/bin/gfortran

```

In this system, we have two gcc modules installed via `spack` package manager, we will attempt to add both modules as compiler instance in buildtest.

```

$ module -t av gcc
/Users/siddiq90/projects/spack/share/spack/lmod/darwin-catalina-x86_64/Core:
gcc/9.3.0-n7p74fd
gcc/10.2.0-37fmsw7

```

Next we run `buildtest config compilers find` which will search all modules based on regular expression and add compilers in their respective group. In this example, buildtest automatically add `gcc/9.2.0-n7p74fd` and `gcc/10.2.0-37fmsw7` modules as compiler instance. Depending on the compiler group, buildtest will apply the compiler wrapper `cc`, `cxx`, `fc` however these can be updated manually. The module section is generated with the module to load. One can further tweak the module behavior along with purging or swap modules.

```

$ buildtest config compilers find
MODULEPATH: /Users/siddiq90/projects/spack/share/spack/lmod/darwin-catalina-x86_64/Core:/
↳usr/local/Cellar/lmod/8.4.12/modulefiles/Darwin:/usr/local/Cellar/lmod/8.4.12/
↳modulefiles/Core
Configuration File: /Users/siddiq90/.buildtest/config.yml
-----
moduletool: lmod
load_default_buildspecs: true
executors:
  local:
    bash:
      description: submit jobs on local machine using bash shell
      shell: bash
    sh:
      description: submit jobs on local machine using sh shell
      shell: sh
    csh:
      description: submit jobs on local machine using csh shell
      shell: csh
    python:
      description: submit jobs on local machine using python shell
      shell: python
compilers:
  find:
    gcc: ^(gcc)
    pgi: ^(pgi)
  compiler:
    gcc:
      builtin_gcc:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++

```

(continues on next page)

(continued from previous page)

```

fc: /usr/local/bin/gfortran
gcc/9.3.0-n7p74fd:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/9.3.0-n7p74fd
    purge: false
gcc/10.2.0-37fmsw7:
  cc: gcc
  cxx: g++
  fc: gfortran
  module:
    load:
      - gcc/10.2.0-37fmsw7
    purge: false

```

Updating settings file: /Users/siddiq90/.buildtest/config.yml

This feature relies on module system (Lmod, environment-modules) to search modulefiles and one must specify **moduletool** property to indicate how buildtest will search modules. If `moduletool: lmod` is set, buildtest will rely on Lmod spider using `Lmodule` API to detect and test all modules. If `moduletool: environment-modules` is set, buildtest will retrieve modules using output of `module -t av`.

3.5.3 Site Examples

Ascent @ OLCF

`Ascent` is a training system for Summit at OLCF, which is using a IBM Load Sharing Facility (LSF) as their batch scheduler. Ascent has two queues **batch** and **test**. To declare LSF executors we define them under `lsf` section within the `executors` section.

The default launcher is `bsub` which can be defined under `defaults`. The `pollinterval` will poll LSF jobs every 10 seconds using `bjobs`. The `pollinterval` accepts a range between **10 - 300** seconds as defined in schema. In order to avoid polling scheduler excessively pick a number that is best suitable for your site

```

system:
  ascent:
    moduletool: lmod
    load_default_buildspecs: false
    executors:
      defaults:
        launcher: bsub
        pollinterval: 10
        max_pend_time: 60
        account: gen014ecpci
      local:
        bash:
          description: submit jobs on local machine using bash shell
          shell: bash

```

(continues on next page)

(continued from previous page)

```

sh:
  description: submit jobs on local machine using sh shell
  shell: sh
csh:
  description: submit jobs on local machine using csh shell
  shell: csh
python:
  description: submit jobs on local machine using python shell
  shell: python
lsf:
  batch:
    queue: batch
  test:
    queue: test
compilers:
  find:
    gcc: ^(gcc)
    pgi: ^(pgi)
    cuda: ^(cuda)
  compiler:
    gcc:
      builtin_gcc:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran

```

JLSE @ ANL

Joint Laboratory for System Evaluation (JLSE) provides a testbed of emerging HPC systems, the default scheduler is Cobalt, this is defined in the `cobalt` section defined in the executor field.

We set default launcher to `qsub` defined with `launcher: qsub`. This is inherited for all batch executors. In each cobalt executor the queue property will specify the queue name to submit job, for instance the executor `yarrow` with `queue: yarrow` will submit job using `qsub -q yarrow` when using this executor.

```

system:
  jlse:
    hostnames:
      - jlselogin*
    moduletool: environment-modules
    load_default_buildspecs: false
    executors:
      defaults:
        launcher: qsub
        pollinterval: 10
        max_pend_time: 300
      local:
        bash:
          description: submit jobs on local machine using bash shell
          shell: bash
        sh:

```

(continues on next page)

(continued from previous page)

```

    description: submit jobs on local machine using sh shell
    shell: sh
csch:
    description: submit jobs on local machine using csh shell
    shell: csh
python:
    description: submit jobs on local machine using python shell
    shell: python
cobalt:
    yarrow:
        queue: yarrow
    yarrow_debug:
        queue: yarrow_debug
    iris:
        queue: iris
    iris_debug:
        queue: iris_debug

```

3.6 Batch Scheduler Support

3.6.1 Slurm

buildtest can submit jobs to [Slurm](#) assuming you have slurm executors defined in your configuration file. The `SlurmExecutor` class is responsible for managing slurm jobs which will perform the following action

1. Check slurm binary `sbatch` and `sacct`.
2. Dispatch Job and acquire job ID using `sacct`.
3. Poll all slurm jobs until all have finished
4. Gather Job results once job is complete via `sacct`.

buildtest will dispatch slurm jobs and poll all jobs until all jobs are complete. If job is in **PENDING** or **RUNNING** state, then buildtest will keep polling at a set interval defined by `pollinterval` setting in buildtest. Once job is not in **PENDING** or **RUNNING** stage, buildtest will gather job results and wait until all jobs have finished.

In this example we have a slurm executor `cori.slurm.knl_debug`, in addition we can specify **#SBATCH** directives using `sbatch` field. The `sbatch` field is a list of string types, buildtest will insert **#SBATCH** directive in front of each value.

Shown below is an example buildspec

```

version: "1.0"
buildspecs:
    slurm_metadata:
        description: Get metadata from compute node when submitting job
        type: script
    executor: cori.slurm.knl_debug
    tags: [jobs]
    sbatch:
        - "-t 00:05"
        - "-N 1"

```

(continues on next page)

(continued from previous page)

```
run: |
    export SLURM_JOB_NAME="firstjob"
    echo "jobname:" $SLURM_JOB_NAME
    echo "slurmdb host:" $SLURMD_NODENAME
    echo "pid:" $SLURM_TASK_PID
    echo "submit host:" $SLURM_SUBMIT_HOST
    echo "nodeid:" $SLURM_NODEID
    echo "partition:" $SLURM_JOB_PARTITION
```

buildtest will add the #SBATCH directives at top of script followed by content in the run section. Shown below is the example test content. Every slurm will insert #SBATCH --job-name, #SBATCH --output and #SBATCH --error line which is determined by the name of the test.

```
#!/bin/bash
#SBATCH -t 00:05
#SBATCH -N 1
#SBATCH --job-name=slurm_metadata
#SBATCH --output=slurm_metadata.out
#SBATCH --error=slurm_metadata.err
export SLURM_JOB_NAME="firstjob"
echo "jobname:" $SLURM_JOB_NAME
echo "slurmdb host:" $SLURMD_NODENAME
echo "pid:" $SLURM_TASK_PID
echo "submit host:" $SLURM_SUBMIT_HOST
echo "nodeid:" $SLURM_NODEID
echo "partition:" $SLURM_JOB_PARTITION
```

The cori.slurm.knl_debug executor in our configuration file is defined as follows

```
system:
  cori:
    executors:
      slurm:
        knl_debug:
          qos: debug
          cluster: cori
          options:
            - -C knl,quad,cache
          description: debug queue on KNL partition
```

With this setting, any buildspec test that use cori.slurm.knl_debug executor will result in the following launch option: sbatch --qos debug --clusters=cori -C knl,quad,cache </path/to/script.sh>.

Unlike the LocalExecutor, the **Run Stage**, will dispatch the slurm job and poll until job is completed. Once job is complete, it will gather the results and terminate. In Run Stage, buildtest will mark test status as N/A because job is submitted to scheduler and pending in queue. In order to get job result, we need to wait until job is complete then we gather results and determine test state. buildtest keeps track of all buildspecs, testscripts to be run and their results. A test using LocalExecutor will run test in **Run Stage** and returncode will be retrieved and status can be calculated immediately. For Slurm Jobs, buildtest dispatches the job and process next job. buildtest will show output of all tests after **Polling Stage** with test results of all tests. A slurm job with exit code 0 will be marked with status PASS.

Shown below is an example build for this test

```

$ buildtest build -b buildsspecs/jobs/metadata.yml
User: siddiq90
Hostname: cori02
Platform: Linux
Current Time: 2021/09/03 10:08:49
buildtest path: /global/homes/s/siddiq90/github/buildtest/bin/buildtest
buildtest version: 0.10.2
python path: /global/homes/s/siddiq90/.conda/envs/buildtest/bin/python
python version: 3.8.8
Test Directory: /global/u1/s/siddiq90/github/buildtest/var/tests
Configuration File: /global/u1/s/siddiq90/.buildtest/config.yml
Command: /global/homes/s/siddiq90/github/buildtest/bin/buildtest build -b buildsspecs/
↪ jobs/metadata.yml

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+-----+
| /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/jobs/metadata.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/global/u1/s/siddiq90/github/buildtest-cori/buildspecs/jobs/metadata.yml: VALID

Total builder objects created: 1
builders: [slurm_metadata/303d1e32]

name          id          description          buildspecs
-----
↪ -----
slurm_metadata 303d1e32 Get metadata from compute node when submitting job /global/u1/
↪ s/siddiq90/github/buildtest-cori/buildspecs/jobs/metadata.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type  | executor          | tags  | testpath
-----+-----+-----+-----+-----+-----
↪ -----
↪ -----

```

(continues on next page)

(continued from previous page)

```

slurm_metadata | 303d1e32 | script | cori.slurm.knl_debug | ['jobs'] | /global/u1/s/
↪siddiq90/github/buildtest/var/tests/cori.slurm.knl_debug/metadata/slurm_metadata/
↪303d1e32/slurm_metadata_build.sh

+-----+
| Stage: Running Test |
+-----+

Launching test: slurm_metadata
Test ID: 303d1e32-52eb-4d77-9a36-04a5143c4cbd
Executor Name: cori.slurm.knl_debug
Running Test: /global/u1/s/siddiq90/github/buildtest/var/tests/cori.slurm.knl_debug/
↪metadata/slurm_metadata/303d1e32/slurm_metadata_build.sh
slurm_metadata/303d1e32 JobID: 46508594 dispatched to scheduler
Polling Jobs in 30 seconds
slurm_metadata/303d1e32: Job 46508594 is complete!
slurm_metadata/303d1e32: Writing output file: /global/u1/s/siddiq90/github/buildtest/var/
↪tests/cori.slurm.knl_debug/metadata/slurm_metadata/303d1e32/slurm_metadata.out
slurm_metadata/303d1e32: Writing error file: /global/u1/s/siddiq90/github/buildtest/var/
↪tests/cori.slurm.knl_debug/metadata/slurm_metadata/303d1e32/slurm_metadata.err

+-----+
| Completed Polled Jobs |
+-----+

name          | id          | executor          | status  | returncode | runtime
-----+-----+-----+-----+-----+-----
slurm_metadata | 303d1e32    | cori.slurm.knl_debug | PASS    | 0          | 30.9923

+-----+
| Stage: Test Summary |
+-----+

name          | id          | executor          | status  | returncode | runtime
-----+-----+-----+-----+-----+-----
slurm_metadata | 303d1e32    | cori.slurm.knl_debug | PASS    | 0          | 30.9923

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_2159tqkz.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /global/homes/s/
↪siddiq90/github/buildtest/buildtest.log

```

The **SlurmExecutor** class is responsible for processing slurm job that may include: dispatch, poll, gather, or cancel job. The SlurmExecutor will gather job metrics via `sacct`.

buildtest can check status based on Slurm Job State, this is defined by `State` field in `sacct`. In next example, we introduce field `slurm_job_state` which is part of `status` field. This field expects one of the following values: `[COMPLETED,`

FAILED, OUT_OF_MEMORY, TIMEOUT] This is an example of simulating fail job by expecting a return code of 1 with job state of FAILED.

```

1 version: "1.0"
2 buildspecs:
3   wall_timeout:
4     type: script
5     executor: cori.slurm.haswell_debug
6     sbatch: [ "-t '00:00:10'", "-n 1" ]
7     description: "This job simulates job timeout by sleeping for 300sec while requesting_
↪5sec"
8     tags: ["jobs", "fail"]
9     run: sleep 180
10    status:
11      slurm_job_state: "TIMEOUT"

```

If we run this test, buildtest will mark this test as PASS because the slurm job state matches with expected result defined by field slurm_job_state. This job will be TIMEOUT because we requested 2 mins while this job will sleep 300sec (5min).

```

(buildtest) siddiq90@cori02> buildtest build -b buildspecs/jobs/fail/timeout.yml
User: siddiq90
Hostname: cori02
Platform: Linux
Current Time: 2021/09/03 13:34:13
buildtest path: /global/homes/s/siddiq90/github/buildtest/bin/buildtest
buildtest version: 0.10.2
python path: /global/homes/s/siddiq90/.conda/envs/buildtest/bin/python
python version: 3.8.8
Test Directory: /global/u1/s/siddiq90/github/buildtest/var/tests
Configuration File: /global/u1/s/siddiq90/.buildtest/config.yml
Command: /global/homes/s/siddiq90/github/buildtest/bin/buildtest build -b buildspecs/
↪jobs/fail/timeout.yml

```

```

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+=====+
| /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/jobs/fail/timeout.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/global/u1/s/siddiq90/github/buildtest-cori/buildspecs/jobs/fail/timeout.yml: VALID

```

(continues on next page)

(continued from previous page)

```

Total builder objects created: 1
builders: [wall_timeout/ae385691]

name          id          description
↪            ↪          ↪
-----
↪            ↪          ↪
↪ -
wall_timeout  ae385691  This job simulates job timeout by sleeping for 300sec while
↪ requesting 5sec  /global/u1/s/siddiq90/github/buildtest-cori/buildspecs/jobs/fail/
↪ timeout.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor                  | tags          | ↪
↪ testpath
-----+-----+-----+-----+-----+-----+
↪
↪
wall_timeout | ae385691 | script | cori.slurm.haswell_debug | ['jobs', 'fail'] | /
↪ global/u1/s/siddiq90/github/buildtest/var/tests/cori.slurm.haswell_debug/timeout/wall_
↪ timeout/ae385691/wall_timeout_build.sh

+-----+
| Stage: Running Test |
+-----+

-----
Launching test: wall_timeout
Test ID: ae385691-9eb4-413c-ac5b-f2be1bcc449e
Executor Name: cori.slurm.haswell_debug
Running Test: /global/u1/s/siddiq90/github/buildtest/var/tests/cori.slurm.haswell_debug/
↪ timeout/wall_timeout/ae385691/wall_timeout_build.sh
wall_timeout/ae385691 JobID: 46518859 dispatched to scheduler
Polling Jobs in 30 seconds

Current Jobs
-----

+-----+-----+-----+-----+-----+-----+
| name   | id     | executor          | jobID  | jobstate | runtime |
+-----+-----+-----+-----+-----+-----+
| wall_timeout | ae385691 | cori.slurm.haswell_debug | 46518859 | RUNNING | 30.38 |
+-----+-----+-----+-----+-----+-----+

Polling Jobs in 30 seconds

```

(continues on next page)

(continued from previous page)

Current Jobs

```

+-----+-----+-----+-----+-----+-----+
| name | id | executor | jobID | jobstate | runtime |
+-----+-----+-----+-----+-----+-----+
| wall_timeout | ae385691 | cori.slurm.haswell_debug | 46518859 | RUNNING | 60.521 |
+-----+-----+-----+-----+-----+-----+

```

Polling Jobs in 30 seconds

wall_timeout/ae385691: Job 46518859 is complete!

wall_timeout/ae385691: Writing output file: /global/u1/s/siddiq90/github/buildtest/var/
↳ tests/cori.slurm.haswell_debug/timeout/wall_timeout/ae385691/wall_timeout.outwall_timeout/ae385691: Writing error file: /global/u1/s/siddiq90/github/buildtest/var/
↳ tests/cori.slurm.haswell_debug/timeout/wall_timeout/ae385691/wall_timeout.err

```

+-----+
| Completed Polled Jobs |
+-----+

```

```

name | id | executor | jobID | jobstate | status |
↳ returncode | runtime
+-----+-----+-----+-----+-----+-----+
↳ -----+-----+
wall_timeout | ae385691 | cori.slurm.haswell_debug | 46518859 | TIMEOUT | PASS |
↳ 0 | 90.6563

```

```

+-----+
| Stage: Test Summary |
+-----+

```

```

name | id | executor | status | returncode_match |
↳ regex_match | runtime_match | returncode | runtime
+-----+-----+-----+-----+-----+-----+
↳ -----+-----+-----+-----+
wall_timeout | ae385691 | cori.slurm.haswell_debug | PASS | False |
↳ False | False | 0 | 90.6563

```

Passed Tests: 1/1 Percentage: 100.000%

Failed Tests: 0/1 Percentage: 0.000%

Writing Logfile to: /tmp/buildtest_yr61l5t9.log

A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /global/homes/s/
↳ siddiq90/github/buildtest/buildtest.logbuildtest marked this test PASS because the jobstate **TIMEOUT** match the value provided by slurm_job_state in the buildspect.

3.6.2 LSF

buildtest can support job submission to [IBM Spectrum LSF](#) if you have defined LSF executors in your configuration file.

The `bsub` property can be used to specify `#BSUB` directive into job script. This example will use the executor `ascent.lsf.batch` executor that was defined in buildtest configuration.

```

1 version: "1.0"
2 buildspecs:
3   hostname:
4     type: script
5     executor: ascent.lsf.batch
6     bsub: [ "-W 10", "-nnodes 1" ]
7
8   run: jsrun hostname

```

The `LSFExecutor` poll jobs and retrieve job state using `bjobs -noheader -o 'stat' <JOBID>`. The `LSFExecutor` will poll job so long as they are in **PEND** or **RUN** state. Once job is not in any of the two states, `LSFExecutor` will gather job results. buildtest will retrieve the following format fields using `bjobs`: `job_name`, `stat`, `user`, `user_group`, `queue`, `proj_name`, `pids`, `exit_code`, `from_host`, `exec_host`, `submit_time`, `start_time`, `finish_time`, `nthreads`, `exec_home`, `exec_cwd`, `output_file`, `error_file` to get job record.

3.6.3 PBS

buildtest can support job submission to [PBS Pro](#) or [OpenPBS](#) scheduler. Assuming you have configured [PBS Executors](#) in your configuration file you can submit jobs to the PBS executor by selecting the appropriate `pbs` executor via `executor` property in `buildspec`. The `#PBS` directives can be specified using `pbs` field which is a list of PBS options that get inserted at top of script. Shown below is an example `buildspec` using the `script` schema.

```

version: "1.0"
buildspecs:
  pbs_sleep:
    type: script
    executor: generic.pbs.workq
    pbs: ["-l nodes=1", "-l walltime=00:02:00"]
    run: sleep 10

```

buildtest will poll PBS jobs using `qstat -x -f -F json <jobID>` until job is finished. Note that we use `-x` option to retrieve finished jobs which is required in order for buildtest to detect job state upon completion.

Shown below is an example build of the `buildspec` using PBS scheduler.

```

[pbsuser@pbs tests]$ python3.7 ./bin/buildtest -c tests/settings/pbs.yml build -b tests/
↳ examples/pbs/sleep.yml --poll-interval=5
User: pbsuser
Hostname: pbs
Platform: Linux
Current Time: 2021/09/03 20:40:24
buildtest path: /tmp/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.10.2
python path: /bin/python
python version: 3.7.0
Test Directory: /tmp/GitHubDesktop/buildtest/var/tests

```

(continues on next page)

(continued from previous page)

```

Configuration File: /tmp/GitHubDesktop/buildtest/tests/settings/pbs.yml
Command: ./bin/buildtest -c tests/settings/pbs.yml build -b tests/examples/pbs/sleep.yml
↳--poll-interval=5

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs          |
+=====+
| /tmp/GitHubDesktop/buildtest/tests/examples/pbs/sleep.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/tmp/GitHubDesktop/buildtest/tests/examples/pbs/sleep.yml: VALID

Total builder objects created: 1
builders: [pbs_sleep/631998a2]

name      id      description      buildsspecs
-----
↳-----
pbs_sleep 631998a2      /tmp/GitHubDesktop/buildtest/tests/examples/pbs/
↳sleep.yml

+-----+
| Stage: Building Test |
+-----+

name      | id      | type   | executor      | tags      | testpath
-----+-----+-----+-----+-----+-----
↳-----
pbs_sleep | 631998a2 | script | generic.pbs.workq |          | /tmp/GitHubDesktop/
↳buildtest/var/tests/generic.pbs.workq/sleep/pbs_sleep/631998a2/pbs_sleep_build.sh

+-----+
| Stage: Running Test |
+-----+

-----
Launching test: pbs_sleep

```

(continues on next page)

(continued from previous page)

```
Test ID: 631998a2-dc7c-4407-9b3f-552be9a11161
Executor Name: generic.pbs.workq
Running Test: /tmp/GitHubDesktop/buildtest/var/tests/generic.pbs.workq/sleep/pbs_sleep/
↳631998a2/pbs_sleep_build.sh
[pbs_sleep] JobID: 394.pbs dispatched to scheduler
Polling Jobs in 5 seconds

Current Jobs
-----

+-----+-----+-----+-----+-----+-----+
| name   | id     | executor      | jobID  | jobstate | runtime |
+-----+-----+-----+-----+-----+-----+
| pbs_sleep | 631998a2 | generic.pbs.workq | 394.pbs | R        | 5.143   |
+-----+-----+-----+-----+-----+-----+

Polling Jobs in 5 seconds
pbs_sleep/631998a2: Job 394.pbs is complete!
pbs_sleep/631998a2: Writing output file: /tmp/GitHubDesktop/buildtest/var/tests/generic.
↳pbs.workq/sleep/pbs_sleep/631998a2/pbs_sleep.o394
pbs_sleep/631998a2: Writing error file: /tmp/GitHubDesktop/buildtest/var/tests/generic.
↳pbs.workq/sleep/pbs_sleep/631998a2/pbs_sleep.e394

+-----+-----+
| Completed Polled Jobs |
+-----+-----+

name      | id      | executor      | jobID  | jobstate | status  |
↳returncode | runtime
-----+-----+-----+-----+-----+-----+
↳--+-----+
pbs_sleep | 631998a2 | generic.pbs.workq | 394.pbs | F        | PASS    |
↳0 | 10.193
-----+-----+

| Stage: Test Summary |
+-----+-----+

name      | id      | executor      | status  | returncode_match | regex_match
↳ | runtime_match | returncode | runtime
-----+-----+-----+-----+-----+-----+
↳--+-----+-----+-----+-----+-----+
pbs_sleep | 631998a2 | generic.pbs.workq | PASS    | N/A             | N/A
↳ | N/A          | 0 | 10.193
-----+-----+-----+-----+-----+-----+

Passed Tests: 1/1 Percentage: 100.000%
Failed Tests: 0/1 Percentage: 0.000%
```

(continues on next page)

(continued from previous page)

```
Writing Logfile to: /tmp/buildtest_moa4gi1x.log
A copy of logfile can be found at $BUILDTEST_ROOT/buildtest.log - /tmp/GitHubDesktop/
↳ buildtest/buildtest.log
```

3.6.4 Cobalt

Cobalt is a job scheduler developed by Argonne National Laboratory that runs on compute resources and IBM BlueGene series. Cobalt resembles PBS in terms of command line interface such as qsub, qacct however they slightly differ in their behavior.

Cobalt support has been tested on JLSE and Theta system. Cobalt directives are specified using #COBALT this can be specified using cobalt property which accepts a list of strings. Shown below is an example using cobalt property.

```
1 version: "1.0"
2 buildspecs:
3   yarrow_hostname:
4     executor: jlse.cobalt.yarrow
5     type: script
6     cobalt: ["-n 1", "--proccount 1", "-t 10"]
7     run: hostname
```

In this example, we allocate 1 node with 1 processor for 10min. This is translated into the following job script.

```
#!/usr/bin/bash
#COBALT -n 1
#COBALT --proccount 1
#COBALT -t 10
#COBALT --jobname yarrow_hostname
source /home/shahzebsiddiqui/buildtest/var/executors/cobalt.yarrow/before_script.sh
hostname
source /home/shahzebsiddiqui/buildtest/var/executors/cobalt.yarrow/after_script.sh
```

When job starts, Cobalt will write a cobalt log file <JOBID>.cobaltlog which is provided by scheduler for troubleshooting. The output and error file are generated once job finishes. Cobalt job progresses through job state starting → pending → running → exiting. buildtest will capture Cobalt job details using qstat -lf <JOBID> and this is updated in the report file.

buildtest will poll job at set interval, where we run qstat --header State <JobID> to check state of job, if job is finished then we gather results. Once job is finished, qstat will not be able to poll job this causes an issue where buildtest can't poll job since qstat will not return anything. This is a transient issue depending on when you poll job, generally at ALCF qstat will not report existing job within 30sec after job is terminated. buildtest will assume if it's able to poll job and is in *exiting* stage that job is complete, if its unable to retrieve this state we check for output and error file. If file exists we assume job is complete and buildtest will gather the results.

buildtest will determine exit code by parsing cobalt log file, the file contains a line such as

```
Thu Nov 05 17:29:30 2020 +0000 (UTC) Info: task completed normally with an exit code of 0; initiating job cleanup and removal
```

qstat has no job record for capturing returncode so buildtest must rely on Cobalt Log file.

3.6.5 Jobs exceeds *max_pend_time*

Recall from *Configuring buildtest* that *max_pend_time* will cancel jobs if job exceed timelimit. buildtest will start a timer for each job right after job submission and keep track of time duration, and if job is in **pending** state and it exceeds *max_pend_time*, then job will be cancelled.

We can also override *max_pend_time* configuration via command line `--max-pend-time`. To demonstrate, here is an example where job was cancelled after job was pending and exceeds *max_pend_time*. Note that cancelled job is not reported in final output nor updated in report hence it won't be present in the report (`buildtest report`). In this example, we only had one test so upon job cancellation we found there was no tests to report hence, buildtest will terminate after run stage.

```
[pbsuser@pbs buildtest]$ python3.7 ./bin/buildtest -c tests/settings/pbs.yml build -b ↵
↵ tests/examples/pbs/hold.yml --poll-interval=3 --max-pend-time=5
User: pbsuser
Hostname: pbs
Platform: Linux
Current Time: 2021/09/03 20:45:34
buildtest path: /tmp/GitHubDesktop/buildtest/bin/buildtest
buildtest version: 0.10.2
python path: /bin/python
python version: 3.7.0
Test Directory: /tmp/GitHubDesktop/buildtest/var/tests
Configuration File: /tmp/GitHubDesktop/buildtest/tests/settings/pbs.yml
Command: ./bin/buildtest -c tests/settings/pbs.yml build -b tests/examples/pbs/hold.yml -
↵ -poll-interval=3 --max-pend-time=5

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
| Discovered Buildsspecs |
+=====+
| /tmp/GitHubDesktop/buildtest/tests/examples/pbs/hold.yml |
+-----+
Discovered Buildsspecs: 1
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 1

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 1
Invalid Buildsspecs: 0
/tmp/GitHubDesktop/buildtest/tests/examples/pbs/hold.yml: VALID

Total builder objects created: 1
builders: [pbs_hold_job/db8014c4]

name          id          description    buildsspecs
```

(continues on next page)

(continued from previous page)

```

-----
↪-----
pbs_hold_job db8014c4 PBS Hold Job /tmp/GitHubDesktop/buildtest/tests/examples/pbs/
↪hold.yml

+-----+
| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags   | testpath
+-----+-----+-----+-----+-----+-----+
↪-----
pbs_hold_job | db8014c4 | script | generic.pbs.workq |        | /tmp/GitHubDesktop/
↪buildtest/var/tests/generic.pbs.workq/hold/pbs_hold_job/db8014c4/pbs_hold_job_build.sh

+-----+
| Stage: Running Test |
+-----+

-----
Launching test: pbs_hold_job
Test ID: db8014c4-547b-487e-9d2e-f3c743adddf9
Executor Name: generic.pbs.workq
Running Test: /tmp/GitHubDesktop/buildtest/var/tests/generic.pbs.workq/hold/pbs_hold_
↪job/db8014c4/pbs_hold_job_build.sh
[pbs_hold_job] JobID: 395.pbs dispatched to scheduler
Polling Jobs in 3 seconds

Current Jobs
-----

+-----+-----+-----+-----+-----+-----+
| name      | id        | executor      | jobID | jobstate | runtime |
+-----+-----+-----+-----+-----+-----+
| pbs_hold_job | db8014c4 | generic.pbs.workq | 395.pbs | H        | 3.167   |
+-----+-----+-----+-----+-----+-----+

Polling Jobs in 3 seconds
pbs_hold_job/db8014c4: Cancelling Job: 395.pbs because job exceeds max pend time: 5 sec_
↪with current pend time of 6.214

Cancelled Jobs: [pbs_hold_job/db8014c4]
Unable to run any tests

```

3.6.6 Cray Burst Buffer & Data Warp

For Cray systems, you may want to stage-in or stage-out into your burst buffer this can be configured using the `#DW` directive. For a list of data warp examples see section on [DataWarp Job Script Commands](#)

In buildtest we support properties BB and DW which is a list of job directives that get inserted as `#BW` and `#DW` into the test script. To demonstrate let's start off with an example where we create a persistent burst buffer named databuffer of size 10GB striped. We access the burst buffer using the `DW` directive. Finally we cd into the databuffer and write a 5GB random file.

Note: BB and DW directives are generated after scheduler directives. The `#BB` comes before `#DW`. buildtest will automatically add the directive `#BB` and `#DW` when using properties BB and DW

```
1 version: "1.0"
2 buildspecs:
3   create_burst_buffer:
4     type: script
5     executor: cori.slurm.debug
6     batch:
7       nodecount: "1"
8       timelimit: "5"
9       cpucount: "1"
10    sbatch: ["-C knl"]
11    description: Create a burst buffer
12    tags: [jobs]
13    BB:
14      - create_persistent name=databuffer capacity=10GB access_mode=striped type=scratch
15    DW:
16      - persistentdw name=databuffer
17    run: |
18      cd $DW_PERSISTENT_STRIPED_databuffer
19      pwd
20      dd if=/dev/urandom of=random.txt bs=1G count=5 iflags=fullblock
21      ls -lh $DW_PERSISTENT_STRIPED_databuffer/
```

Next we run this test and inspect the generated test we will see that `#BB` and `#DW` directives are inserted after the scheduler directives

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=5
#SBATCH --ntasks=1
#SBATCH --job-name=create_burst_buffer
#SBATCH --output=create_burst_buffer.out
#SBATCH --error=create_burst_buffer.err
#BB create_persistent name=databuffer capacity=10GB access_mode=striped type=scratch
#DW persistentdw name=databuffer
cd $DW_PERSISTENT_STRIPED_databuffer
pwd
dd if=/dev/urandom of=random.txt bs=1G count=5 iflag=fullblock
ls -lh $DW_PERSISTENT_STRIPED_databuffer
```

We can confirm there is an active burst buffer by running the following

```
$ scontrol show burst | grep databuffer
Name=databuffer CreateTime=2020-10-29T13:06:21 Pool=wlm_pool Size=20624MiB
↪State=allocated UserID=siddiq90(92503)
```

3.7 Build and Test Process

The *buildtest build* command is responsible for building and running tests. Every builds spec goes through a pipeline that discovers builds specs, validates the builds spec and builds and runs the test. The builds spec must go through each stage of the pipeline, if it fails in one of the stage, the builds spec will be ignored.



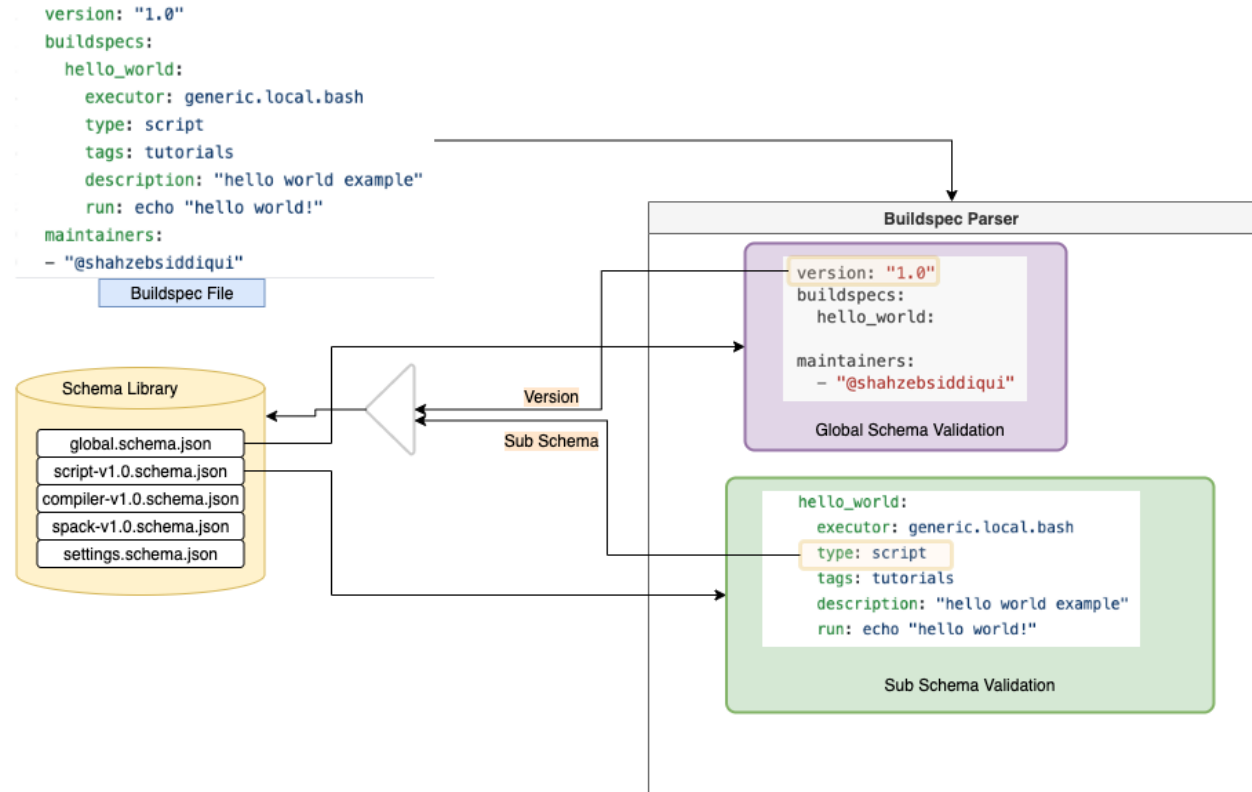
3.7.1 Discover Builds specs

buildtest will discover builds specs based on command line arguments since you can build by file, directory, executor, or tags. In **discover** stage, buildtest will detect builds specs which is discussed in *Discover Builds specs*.

For every discovered builds specs, buildtest will validate the builds specs in the **parse** stage which is performed using *jsonschema.validate* library. The parser will validate every builds spec with the global schema named *global.schema.json* and one of the sub-schemas, check *parsing builds specs* section for more details.

3.7.2 Parse Builds specs

A builds spec file may contain one or more test sections specified via *buildspec* field. Each test is validated by a sub-schema specified by *type* field. buildtest will validate the builds spec with global schema first followed by sub-schema by using the *version* field to look up the schema version for sub-schema. buildtest will look up the schema from its schema library and validate the test section *hello_world* with schema *script-v1.0.schema.json*.



Buildspecs will be ignored if it fails validation process for instance you may have an *Invalid Buildspecs*. Invalid buildspecs won't be sent to **build** stage since we can't reliably build a test-script.

3.7.3 Building Buildspecs

buildtest will send all valid buildspecs to **build** phase which is responsible for building a shell-script from the buildspec file. In this stage, we create a **Builder** object that is an instance of *BuilderBase* class that is a base class for building a buildspec. There is a sub-class for *BuilderBase* class such as *ScriptBuilder* and *CompilerBuilder* that implements how to build a test-script based on the sub-schema selection (`type: compiler`).

During build phase, there are additional checks on buildspecs to ensure we can generate a test-script. In the event of failure, buildtest will raise an exception and buildspec will be ignored. The ignored buildspecs are not sent to **run** stage

3.7.4 Running Buildspecs

In this stage, we run the test based on *executors* defined in configuration file. buildtest will select the executor defined by `executor` property in buildspec which is responsible for running the test. There is a *BaseExecutor* that is a base-class for all executors. We have sub-class for each executor type (Local, Slurm, Cobalt, PBS, Cobalt). In this stage, we run the test and get output, error, returncode and detect status of test (PASS, FAIL). If test is run via scheduler, we submit job to scheduler and poll jobID until it is finished.

Upon completion of test, we update the **Builder** object with the test results which is written to report file.

3.8 Additional Features

3.8.1 Accessing build history (buildtest history)

Note: `buildtest hy` is an alias for `buildtest history` command.

buildtest keeps track of all builds (`buildtest build`) that can be retrieved using `buildtest history` command which can be useful when you want to analyze or troubleshoot past builds. The *buildtest history* command comes with two subcommands `buildtest history list` and `buildtest history query`.

If you want to list all builds you should run **buildtest history list** which will report a table style format of all builds with corresponding build ID to differentiate each build. Shown below is an example output. The build IDs start at **0** and increment as you run **buildtest build** command.

```
$ buildtest history list
```

id	hostname	user	system	date	pass_tests	fail_tests	total_tests	pass_rate	fail_rate	command
0	build-14673784-project-280831-buildtest	docs	generic	2021/09/09 15:54:50	1	0	1	100	0	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/buildtest build -b /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/vars.yml
1	build-14673784-project-280831-buildtest	docs	generic	2021/09/09 15:54:51	2	2	4	50	50	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/buildtest build -b tutorials/pass_returncode.yml
2	build-14673784-project-280831-buildtest	docs	generic	2021/09/09 15:54:51	1	1	2	50	50	/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/buildtest build -b tutorials/status_regex.yml

(continues on next page)

(continued from previous page)

```

+-----+
|      3 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
15:54:59 |      3 |      2 |      5 |      60 |      40 | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
buildtest build -b tutorials/runtime_status_test.yml
|
+-----+
|      4 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
15:54:59 |      2 |      0 |      2 |     100 |      0 | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
buildtest build -b tutorials/shebang.yml
|
+-----+
|      5 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
15:55:00 |      1 |      0 |      1 |     100 |      0 | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
buildtest build -b tutorials/skip_tests.yml
|
+-----+
|      6 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
15:55:00 |      1 |      0 |      1 |     100 |      0 | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
buildtest build -b tutorials/metrics_regex.yml
|
+-----+
|      7 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
15:55:01 |      1 |      0 |      1 |     100 |      0 | /
home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
buildtest build -b tutorials/metrics_variable.yml
|

```

(continues on next page)

(continued from previous page)

```
|      8 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
→15:55:01 |          2 |          0 |          2 |          100 |          0 | /
→home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
→buildtest build -b tutorials/executor_regex_script.yml
→
+-----+-----+-----+-----+-----+-----+-----+
→-+-----+-----+-----+-----+-----+-----+-----+
→
→
→-----+
|      9 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
→15:55:02 |          2 |          0 |          2 |          100 |          0 | /
→home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
→buildtest build -b tutorials/script/multiple_executors.yml
→
+-----+-----+-----+-----+-----+-----+-----+
→-+-----+-----+-----+-----+-----+-----+-----+
→
→
→-----+
|     10 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
→15:55:02 |          2 |          0 |          2 |          100 |          0 | /
→home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
→buildtest build -b tutorials/script/executor_scheduler.yml
→
+-----+-----+-----+-----+-----+-----+-----+
→-+-----+-----+-----+-----+-----+-----+-----+
→
→
→-----+
|     11 | build-14673784-project-280831-buildtest | docs | generic | 2021/09/09
→15:55:03 |          1 |          1 |          2 |          50 |          50 | /
→home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/bin/
→buildtest build -b tutorials/script/status_by_executors.yml
→
+-----+-----+-----+-----+-----+-----+-----+
→-+-----+-----+-----+-----+-----+-----+-----+
→
→
→-----+
```

The `buildtest history query` command is particularly useful when you want to inspect a particular build. This command expects a *Build Identifier* which can be found by inspecting output column `id` in *buildtest history list*.

Shown below is an output of build ID 0 which reports relevant detail for the build such as input command, username, hostname, platform, date, etc...

```
$ buildtest history query 0
{
  "command": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
→0/bin/buildtest build -b /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
→checkouts/v0.11.0/tutorials/vars.yml",
  "user": "docs",
```

(continues on next page)

(continued from previous page)

```

"hostname": "build-14673784-project-280831-buildtest",
"platform": "Linux",
"date": "2021/09/09 15:54:50",
"buildtest": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/bin/buildtest",
"python": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↪ python",
"python_version": "3.7.9",
"testdir": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↪ 0/var/tests",
"configuration": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪ v0.11.0/buildtest/settings/config.yml",
"system": "generic",
"logpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.
↪ 0/var/.history/0/buildtest_8xn_louq.log",
"invalid_buildspecs": [],
"buildspecs": {
  "detected": [
    "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪ tutorials/vars.yml"
  ],
  "included": [
    "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪ tutorials/vars.yml"
  ],
  "excluded": []
},
"test_summary": {
  "pass": "1",
  "fail": "0",
  "total": "1",
  "pass_rate": "100.000",
  "fail_rate": "0.000"
},
"builders": {
  "98b07db8-9b3e-4ff0-b526-1ebcebbc3a1e": {
    "name": "variables_bash",
    "buildspec": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪ v0.11.0/tutorials/vars.yml",
    "tags": [
      "tutorials"
    ],
    "executors": "generic.local.bash",
    "state": "PASS",
    "returncode": 0,
    "runtime": 0.012987,
    "testpath": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪ v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash.sh",
    "errfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪ v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash.err",
    "outfile": "/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↪ v0.11.0/var/tests/generic.local.bash/vars/variables_bash/98b07db8/variables_bash.out"
  }
}

```

(continues on next page)

(continued from previous page)

```
}
}
}
```

If you want to see all available build IDs, you can use the following command. The `-t` is terse format and `--no-header` will omit the headers for each column and pipe the output to `cut` to extract the first column which corresponds to build IDs.

```
$ buildtest hy list -t --no-header | cut -f 1 -d '|'
0
1
2
3
4
5
6
7
8
9
10
11
```

buildtest has tab complete on `buildtest history query` which reports a list of build IDs which is another way to see available IDs to query.

```
$ buildtest history query
0 1 10 11 12 13 14 15 16 17 18 19 2 20 21 22 23 3 4 5 6 7
↩8 9
```

If you want to see logfile for build ID 0 you can use `--log` option to see logfile in an editor as follows:

```
buildtest history query 0 --log
```

3.8.2 Accessing buildtest documentation

We provide two command line options to access main documentation and schema docs. This will open a browser on your machine.

To access [buildtest docs](#) you can run:

```
buildtest docs
```

To access [schema docs](#) you can run:

```
buildtest schemadocs
```

3.8.3 Color Mode

buildtest will display output in color by default which can be configured on command line via `buildtest --color [on|off]` or via environment variable `BUILDTEST_COLOR`. You can disable color output via command argument `--color off` or environment `BUILDTEST_COLOR=False`. If `--color on` is set with `BUILDTEST_COLOR=False`, the value of environment variable will be honored.

3.8.4 CDASH Integration (buildtest cdash)

The `buildtest cdash` command is responsible for uploading tests to CDASH server. You will need to specify *Configuring test directory* in your configuration file. Shown below is the command usage.

```
$ buildtest cdash --help
usage: buildtest [options] [COMMANDS] cdash [-h] ...
```

optional arguments:

`-h, --help` show this help message and exit

subcommands:

`buildtest CDASH integration`

```
view      Open CDASH project in webbrowser
upload    Upload Test to CDASH server
```

The `buildtest cdash upload` command is responsible for uploading all tests in `report.json` into CDASH. You must specify a buildname when using **buildtest cdash upload** in this example we will specify a buildname called *tutorials*:

```
$ buildtest cdash upload tutorials
Reading configuration file: /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/
↳ settings/config.yml
Reading report file: /Users/siddiq90/.buildtest/report.json
build name: tutorials
site: generic
stamp: 20210428-1512-Experimental
MD5SUM: d7651cb3fbdd19298b0188c441704c3a
PUT STATUS: 200
You can view the results at: https://my.cdash.org//viewTest.php?buildid=2004360
```

We can see the output of these tests in CDASH if we go to url <https://my.cdash.org//viewTest.php?buildid=2004360>

Testing started on 2021-04-28 19:12:13

Site Name: generic
Build Name: tutorials
Total time: 10s 500ms

3 passed, 2 failed, 0 not run, 0 missing.

Name	Status	Time	Details	Labels	Summary	Description
timelimit_min_max	Passed	2s 220ms		tutorials	Stable	Run a sleep job for 2 seconds and test pass if its within 1.0-4.0sec
timelimit_min	Passed	2s 60ms		tutorials	Stable	Run a sleep job for 2 seconds and test pass if its exceeds min time of 1.0 sec
timelimit_max	Passed	2s 60ms		tutorials	Stable	Run a sleep job for 2 seconds and test pass if it's within max time: 5.0 sec
timelimit_min_fail	Failed	2s 70ms		tutorials	Broken	This test fails because it runs less than minimum of 1.0 second
timelimit_max_fail	Failed	2s 60ms		tutorials	Broken	This test fails because it exceeds maxtime of 1.0 second

Download Table as CSV File

Kitware

CDash v3.0.3-6-g9447bda © Kitware Report problems | View as JSON | 0.27s (0.1s)
Current Testing Day: 2021-04-28 | Server at 01:00:02C

By default buildtest will read the report file in your **\$HOME/.buildtest/report.json**, we can specify an alternate report file. First let's see the available help options for `buildtest cdash upload`.

```
$ buildtest cdash upload --help
usage: buildtest [options] [COMMANDS] cdash upload [-h] [-r REPORT] [--site SITE]
↪ buildname

positional arguments:
  buildname              Specify Build Name reported in CDASH

optional arguments:
  -h, --help            show this help message and exit
  -r REPORT, --report REPORT
                        Path to report file to upload test results
  --site SITE           Specify site name reported in CDASH
```

We can pass an alternate report file using `-r` option when uploading tests to CDASH. This can be useful if you want to map test results to different buildnames in CDASH perhaps running a different subset of tests via `buildtest build --tags` and upload the test results with different buildname assuming you have different paths to report file.

Let's say we want to build all python tests using tags and store them in a report file which we want to push to CDASH with buildgroup name python we can do that as follows

```
$ buildtest build --tags python -r $BUILDTEST_ROOT/python.json
User: docs
Hostname: build-14673784-project-280831-buildtest
Platform: Linux
Current Time: 2021/09/09 15:55:07
buildtest path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/bin/buildtest
buildtest version: 0.11.0
python path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↪ python
python version: 3.7.9
Test Directory: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↪ 11.0/var/tests
Configuration File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪ checkouts/v0.11.0/buildtest/settings/config.yml
Command: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↪ bin/buildtest build --tags python -r /home/docs/checkouts/readthedocs.org/user_builds/
↪ buildtest/checkouts/v0.11.0/python.json

+-----+
| Stage: Discovering Buildsspecs |
+-----+

+-----+
↪ | Discovered Buildsspecs |
↪ |
+-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↪ python-hello.yml |
+-----+
```

(continues on next page)

(continued from previous page)

```
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml |
+-----+
↳-----+
Discovered Buildsspecs: 2
Excluded Buildsspecs: 0
Detected Buildsspecs after exclusion: 2

BREAKDOWN OF BUILDSPECS BY TAGS
-----
Detected Tag Names: ['python']
+-----+
↳-----+
| python
↳
+=====+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml |
+-----+
↳-----+
| /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml |
+-----+
↳-----+

+-----+
| Stage: Parsing Buildsspecs |
+-----+

Valid Buildsspecs: 2
Invalid Buildsspecs: 0
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-hello.yml: VALID
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/
↳python-shell.yml: VALID

Total builder objects created: 2
builders: [python_hello/66936186, circle_area/8f494a0c]

name          id          description          buildsspecs
-----
↳-----
python_hello  66936186  Hello World python          /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-hello.yml
circle_area   8f494a0c  Calculate circle of area given a radius /home/docs/checkouts/
↳readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/tutorials/python-shell.yml
+-----+
```

(continues on next page)

(continued from previous page)

```

| Stage: Building Test |
+-----+

name          | id          | type   | executor          | tags          |
↳testpath
+-----+
↳
↳
python_hello | 66936186 | script | generic.local.bash | python        | /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/python-hello/python_hello/66936186/python_hello_build.sh
circle_area  | 8f494a0c | script | generic.local.python | ['tutorials', 'python'] | /
↳home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.python/python-shell/circle_area/8f494a0c/circle_area_build.sh
+-----+
| Stage: Running Test |
+-----+

circle_area/8f494a0c: completed with returncode: 0
circle_area/8f494a0c: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/8f494a0c/circle_area.out
circle_area/8f494a0c: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.python/python-shell/circle_
↳area/8f494a0c/circle_area.err
python_hello/66936186: completed with returncode: 0
python_hello/66936186: Writing output file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
↳hello/66936186/python_hello.out
python_hello/66936186: Writing error file: /home/docs/checkouts/readthedocs.org/user_
↳builds/buildtest/checkouts/v0.11.0/var/tests/generic.local.bash/python-hello/python_
↳hello/66936186/python_hello.err

-----
Launching test: python_hello
Test ID: 66936186-97cf-49ff-9f4b-25e6d4a7cb82
Executor Name: generic.local.bash
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.bash/python-hello/python_hello/66936186/python_hello_
↳build.sh

-----
Launching test: circle_area
Test ID: 8f494a0c-c597-4a6b-93e9-07a9f7234ec9
Executor Name: generic.local.python
Running Test: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳11.0/var/tests/generic.local.python/python-shell/circle_area/8f494a0c/circle_area_
↳build.sh

+-----+
| Stage: Test Summary |
+-----+

```

(continues on next page)

(continued from previous page)

name	id	executor	status	returncode_match	regex_
↪match	↪runtime_match	↪returncode	↪runtime		
-----+-----+-----+-----+-----+-----					
circle_area	8f494a0c	generic.local.python	PASS	N/A	N/A
↪	N/A	0	0.11377		
python_hello	66936186	generic.local.bash	PASS	N/A	N/A
↪	N/A	0	0.122556		
-----+-----+-----+-----+-----+-----					
Passed Tests: 2/2 Percentage: 100.000%					
Failed Tests: 0/2 Percentage: 0.000%					
Writing Logfile to: /tmp/buildtest_k8x48zrz.log					
A copy of logfile can be found at \$BUILDTEST_ROOT/buildtest.log - /home/docs/checkouts/					
↪readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest.log					

Next we upload the tests using the `-r` option to specify the report file

```
$ buildtest cdash upload -r $BUILDTEST_ROOT/python.json python
Reading configuration file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/buildtest/settings/config.yml
Reading report file: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↪checkouts/v0.11.0/python.json
build name: python
site: generic
stamp: 20210909-1555-Experimental
MD5SUM: afc72529503d490df26855e3012a78c3
PUT STATUS: 200
You can view the results at: https://my.cdash.org//viewTest.php?buildid=2064112
```

The `buildtest cdash view` command can be used to open CDASH project in a web browser using the command line. This feature assumes you have set the CDASH setting in your configuration file.

3.8.5 Cleaning buildtest files (`buildtest clean`)

The `buildtest clean` command can be used to remove files generated by buildtest such as test files, report files, builds spec cache, and history files. You will be prompted for response to clean up files for confirmation. If you want to avoid user response you can use `buildtest clean -y` to accept confirmation for all prompts and buildtest will remove the files.

```
$ buildtest clean
Remove Test Directory /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests (y/n)↪
↪[default: y]
Remove Report File /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/report.json (y/
↪n) [default: y]
Remove History Directory /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/.history↪
↪(y/n) [default: y]
Remove Builds spec Cache /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/buildspecs/
↪cache.json (y/n) [default: y]
```

(continues on next page)

(continued from previous page)

```

=====> Remove Test Directory
=====> Removing Report File
=====> Removing History Directory
=====> Removing buildspect cache

```

3.8.6 Changing Directories (buildtest cd)

The `buildtest cd` command can be used to change directory to root of test given a test name. The change will be applied to your shell upon completion of command. Let's assume we want to change directory to root of test `exit1_pass` we can do this as follows:

```

$ buildtest cd exit1_pass
Changing directory to root of test: exit1_pass/8c4b6ac9-e94e-40d9-8d96-7aaa3a5d3723

$ pwd
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.sh/pass_
↳returncode/exit1_pass/8c4b6ac9

```

In this previous example, `buildtest` will use the **latest** run for test `exit1_pass` and switch directory to root of test.

We can confirm this directory is from the latest run by running the following command. The `testroot` is a property in the report table that can be fetch via `--format` field. The `--latest` option will fetch the latest run for the test.

```

$ buildtest report --latest --filter name=exit1_pass --format testroot --terse --no-
↳header
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.sh/pass_
↳returncode/exit1_pass/8c4b6ac9

```

If you switch `cd` into a particular build you can specify the name followed by backslash and name of test ID. In this example below, we will specify test name `kernel_swapusage/1fa` and `buildtest` will attempt to find first record that starts with the test ID and switch directory to root of test.

```

$ buildtest cd kernel_swapusage/1fa
Changing directory to root of test: kernel_swapusage/1fa21875-b099-41b6-8bc7-30e0d2dcc13b

$ pwd
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.bash/kernel_
↳state/kernel_swapusage/1fa21875

```

3.8.7 Get Path for Test (buildtest path)

The `buildtest path` command is used to display path attributes for a test that is available in the test report. Shown below are available options for **buildtest path**

```

$ buildtest path -h
usage: buildtest [options] [COMMANDS] path [-h] [-t | -o | -e | -b | --stagedir] name

positional arguments:
  name                  Name of test

```

(continues on next page)

(continued from previous page)

optional arguments:

```
-h, --help          show this help message and exit
-t, --testpath      Show path to test script
-o, --outfile       Show path to output file
-e, --errfile       Show path to error file
-b, --buildscript   Show path to build script
--stagedir          Show path to stage directory
```

If you want to fetch the last run for any given test you can specify the name of the test as follows: `buildtest path <name>`. We can specify a test ID for a test by separating the name and test ID with backslash character (/) as follows: `buildtest path <name>/<ID>`

If you don't specify any option you will get root of test. In this example, we will retrieve `testroot` for test `variables_bash` which is a property of the test found in the report file.

```
$ buildtest path variables_bash
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/vars/variables_bash/98b07db8
```

You can get path to testscript via `-t` option as show below

```
$ buildtest path -t variables_bash
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/tests/
↳generic.local.bash/vars/variables_bash/98b07db8/variables_bash.sh
```

If you want to see content of output file, you can use `-o` option with `cat` command as follows:

```
$ cat $(buildtest path -o variables_bash)
1+2=3
this is a literal string ':'
'singlequote'
"doublequote"
current user: docs
number of files: 4
```

In this next example we will query test `circle_area` with build ID `aaa` and buildtest will find the first match record that starts with this record and resolves to `aaaa622d` which is the short ID of test.

```
# query testroot for circle_area with build ID "aaa"
$ buildtest path circle_area/aaa
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/python-
↳shell/circle_area/aaaa622d

# query testroot for latest run of circle_area
$ buildtest path circle_area
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/python-
↳shell/circle_area/fc221b84
```

We have setup mutual exclusion to avoid user from passing two option at same time. If you do run into this situation you will get the following error.

```
$ buildtest path -o -e variables_bash
usage: buildtest [options] [COMMANDS] path [-h] [-t | -o | -e | -b | --stagedir] name
buildtest [options] [COMMANDS] path: error: argument -e/--errfile: not allowed with
↳argument -o/--outfile
```

(continues on next page)

(continued from previous page)

If you specify an invalid test name or buildtest can't find the test id, then buildtest will print list of available test names with IDs.

3.8.8 Command Line Interface to buildtest configuration

Once you have implemented your buildtest configuration, you can query the configuration details using `buildtest config` command. Shown below is the command usage.

```
$ buildtest config --help
usage: buildtest [options] [COMMANDS] config [-h] ...

optional arguments:
  -h, --help  show this help message and exit

subcommands:
  Query information from buildtest configuration file

  compilers      Search compilers
  executors       Query executors from buildtest configuration
  summary        Provide summary of buildtest settings.
  systems        List all available systems
  validate       Validate buildtest settings file with schema.
  view          View Buildtest Configuration File
```

Note: `buildtest cg` is an alias for `buildtest config` command.

Validate buildtest configuration (`buildtest config validate`)

First thing you should do once you implement your configuration file is to make sure your configuration is valid with the schema. This can be achieved by running `buildtest config validate`. When you invoke this command, buildtest will load the configuration and attempt to validate the file with schema `settings.schema.json`. If validation is successful you will get the following message:

```
$ buildtest config validate
/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/buildtest/
↪ settings/config.yml is valid
```

Note: If you defined a user setting (`~/.buildtest/config.yml`) buildtest will validate this file instead of default one.

If there is an error during validation, the output from `jsonschema.exceptions.ValidationError` will be displayed in terminal. For example the error below indicates that `moduletool` property was expecting one of the values [`environment-modules`, `lmod`, `N/A`] but it recieved a value of `none`:

```
$ buildtest config validate
Traceback (most recent call last):
  File "/Users/siddiq90/Documents/buildtest/bin/buildtest", line 17, in <module>
    buildtest.main.main()
  File "/Users/siddiq90/Documents/buildtest/buildtest/main.py", line 39, in main
    buildtest_configuration = check_settings(settings_file, retrieve_settings=True)
  File "/Users/siddiq90/Documents/buildtest/buildtest/config.py", line 41, in check_
↪ settings
    validate(instance=user_schema, schema=config_schema)
  File "/Users/siddiq90/.local/share/virtualenvs/buildtest-1gHVG2Pd/lib/python3.7/site-
↪ packages/jsonschema/validators.py", line 934, in validate
    raise error
jsonschema.exceptions.ValidationError: 'none' is not one of ['environment-modules', 'lmod
↪ ', 'N/A']

Failed validating 'enum' in schema['properties']['moduletool']:
    {'description': 'Specify modules tool used for interacting with '
        '`module` command. ',
      'enum': ['environment-modules', 'lmod', 'N/A'],
      'type': 'string'}

On instance['moduletool']:
    'none'
```

View buildtest configuration (buildtest config view)

If you want to view buildtest configuration you can run `buildtest config view` which will print content of buildtest configuration.

```
$ buildtest config view
hostnames:
- .*
description: Generic System
moduletool: N/A
load_default_buildspecs: true
executors:
  local:
    bash:
      description: submit jobs on local machine using bash shell
      shell: bash
    sh:
      description: submit jobs on local machine using sh shell
      shell: sh
    csh:
      description: submit jobs on local machine using csh shell
      shell: csh
    zsh:
      description: submit jobs on local machine using zsh shell
      shell: zsh
      disable: true
  python:
    description: submit jobs on local machine using python shell
```

(continues on next page)

(continued from previous page)

```

    shell: python
compilers:
  compiler:
    gcc:
      builtin_gcc:
        cc: gcc
        fc: gfortran
        cxx: g++
cdash:
  url: https://my.cdash.org/
  project: buildtest
  site: generic
  buildname: tutorials

```

Settings File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
 ↪11.0/buildtest/settings/config.yml

Note: buildtest config view will display contents of user buildtest settings ~/.buildtest/config.yml if found, otherwise it will display the default configuration

View Executors (buildtest config executors)

You can use the command `buildtest config executors` to view executors from buildtest configuration file. Shown below is the command usage

```

$ buildtest config executors --help
usage: buildtest [options] [COMMANDS] config executors [-h] [-j | -y | -d | -i]

optional arguments:
  -h, --help            show this help message and exit
  -j, --json            View executor in JSON format
  -y, --yaml            View executors in YAML format
  -d, --disabled        Show disabled executors
  -i, --invalid         Show invalid executors

```

You can run `buildtest config executors` without any options and it will report a list of named executors that you would reference in builds spec using the executor property. If you prefer json or yaml format you can use `--json` or `--yaml` option.

```

$ buildtest config executors
generic.local.bash
generic.local.sh
generic.local.csh
generic.local.python

```

View Registered Systems

Your buildtest configuration may compose of one or more systems since you can define multiple systems in a single configuration file to run buildtest for different HPC clusters. You can use `buildtest config systems` to report all system details defined in your configuration file. In this example below we should the generic system. If you have multiple entries, you will see one entry per system record.

```
$ buildtest config systems
+-----+-----+-----+-----+
| system | description | hostnames | moduletool |
+-----+-----+-----+-----+
| generic | Generic System | ['.*'] | N/A |
+-----+-----+-----+-----+
```

Configuration Summary

You can get a summary of buildtest using `buildtest config summary`, this will display information from several sources into one single command along.

```
$ buildtest config summary
buildtest version: 0.11.0
buildtest Path: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.
↳ 11.0/bin/buildtest

Machine Details
-----
Operating System: ubuntu
Hostname: build-14673784-project-280831-buildtest
Machine: x86_64
Processor: x86_64
Python Path /home/docs/checkouts/readthedocs.org/user_builds/buildtest/envs/v0.11.0/bin/
↳ python
Python Version: 3.7.9
User: docs

Buildtest Settings
-----
Buildtest Settings: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/
↳ v0.11.0/buildtest/settings/config.yml
Executors: ['local.bash', 'local.sh', 'local.csh', 'local.zsh', 'local.python']
Buildspec Cache File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/
↳ checkouts/v0.11.0/var/buildspecs/cache.json

Buildtest Schemas
-----
Available Schemas: ['script-v1.0.schema.json', 'compiler-v1.0.schema.json', 'global.
↳ schema.json', 'settings.schema.json']
```

Example Configurations

buildtest provides a few example configurations for configuring buildtest this can be retrieved by running `buildtest schema -n settings.schema.json --examples` or short option `(-e)`, which will validate each example with schema file `settings.schema.json`.

```
$ buildtest schema -n settings.schema.json -e
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/settings.schema.json/valid/slurm-example.yml
```

```
system:
  generic:
    hostnames: ['.*']

    moduletool: lmod
    load_default_buildspecs: True
    buildspec_roots:
      - $HOME/buildtest-cori
    testdir: /tmp/buildtest
    executors:
      defaults:
        pollinterval: 20
        launcher: sbatch
        max_pend_time: 30
        account: admin
      slurm:
        normal:
          options: ["-C haswell"]
          qos: normal
          before_script: |
            time
            echo "commands run before job"

    compilers:
      compiler:
        gcc:
          default:
            cc: /usr/bin/gcc
            cxx: /usr/bin/g++
            fc: /usr/bin/gfortran
```

```
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/settings.schema.json/valid/cobalt-example.yml
```

```
system:
  generic:
    hostnames: ['.*']

    moduletool: lmod
    load_default_buildspecs: True
    executors:
      defaults:
        launcher: qsub
        max_pend_time: 30
```

(continues on next page)

(continued from previous page)

```

cobalt:
  knl:
    queue: knl

  haswell:
    queue: haswell

compilers:
  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/settings.schema.json/valid/pbs-example.yml

```

system:
  generic:
    hostnames: ['.*']

  moduletool: N/A
  load_default_buildspecs: True
  executors:
    defaults:
      pollinterval: 10
      launcher: qsub
      max_pend_time: 30
    pbs:
      workq:
        queue: workq
  compilers:
    compiler:
      gcc:
        default:
          cc: /usr/bin/gcc
          cxx: /usr/bin/g++
          fc: /usr/bin/gfortran

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/settings.schema.json/valid/lsf-example.yml

```

system:
  generic:
    hostnames: ['.*']

  moduletool: lmod
  load_default_buildspecs: False
  executors:
    defaults:
      pollinterval: 10
      launcher: bsub

```

(continues on next page)

(continued from previous page)

```

    max_pend_time: 45
lsf:
  batch:
    description: "LSF Executor name 'batch' that submits jobs to 'batch' queue"
    queue: batch
    account: developer
    options: ["-W 20"]
    before_script: |
      time
      echo "commands run before job"
  test:
    description: "LSF Executor name 'test' that submits jobs to 'test' queue"
    launcher: bsub
    queue: test
    account: qa
    options: ["-W 20"]
compilers:
  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/settings.schema.json/valid/local-executor.yml

```

```

system:
  generic:
    hostnames: ['.*']

    logdir: $BUILDTEST_ROOT/logs
    testdir: $BUILDTEST_ROOT/tests

    moduletool: N/A
    load_default_buildspecs: False
    cdash:
      url: https://my.cdash.org
      project: buildtest
      site: laptop
    processor:
      numcpus: 8
      cores: 4
      threads_per_core: 2
      sockets: 1
      model: "Intel(R) Core(TM) i7-8569U CPU @ 2.80GHz"
    executors:
      local:
        bash:
          description: submit jobs on local machine using bash shell
          shell: bash
          before_script: |
            time

```

(continues on next page)

(continued from previous page)

```

    echo "commands run before job"

sh:
  description: submit jobs on local machine using sh shell
  shell: sh

csh:
  description: submit jobs on local machine using csh shell
  shell: csh -x

tcsh:
  description: submit jobs on local machine using tcsh shell
  shell: /bin/tcsh

zsh:
  description: submit jobs on local machine using zsh shell
  shell: /bin/zsh

python:
  description: submit jobs on local machine using python shell
  shell: python

compilers:
  find:
    gcc: "^(gcc|GCC|PrgEnv-gnu)"
    intel: "^(intel|Intel|PrgEnv-intel)"
    cray: "^(cray|PrgEnv-cray)"
    clang: "^(clang|Clang)"
    cuda: "^(cuda|CUDA)"
    pgi: "^(pgi|PGI|PrgEnv-pgi)"

  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran
      gcc@7.2.0:
        cc: 'cc'
        cxx: 'c++'
        fc: 'fc'
      module:
        load:
          - gcc/7.2.0
    intel:
      intel@2019:
        cc: 'icc'
        cxx: 'icpc'
        fc: 'ifort'
      module:
        purge: True
        load:

```

(continues on next page)

(continued from previous page)

```

    - gcc/7.2.0
    - intel/2019
cray:
  craype@2.6.2:
    cc: 'cc'
    cxx: 'CC'
    fc: 'fc'
    module:
      load: [craype/2.6.2]
      swap: [PrgEnv-gnu, PrgEnv-cray]

  clang:
    clang@12.0.0:
      cc: 'clang'
      cxx: 'clang++'
      fc: 'None'
      module:
        load: [clang/12.0]
  cuda:
    cuda@11.0:
      cc: 'nvcc'
      cxx: 'nvcc'
      fc: 'None'
      module:
        load: [cuda/11.0]
  pgi:
    pgi@18.0:
      cc: 'pgcc'
      cxx: 'pgc++'
      fc: 'pgfortran'
      module:
        load: [pgi/18.0]

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/settings.schema.json/valid/combined_executor.yml

```

system:
  generic:
    hostnames: ['. *']

    moduletool: N/A
    load_default_buildspecs: True
  executors:
    local:
      bash:
        description: submit jobs on local machine
        shell: bash -v

    slurm:
      haswell:
        launcher: sbatch
        options:

```

(continues on next page)

(continued from previous page)

```

    - "-p haswell"
    - "-t 00:10"

lsf:
  batch:
    launcher: bsub
    queue: batch
    options:
      - "-q batch"
      - "-t 00:10"
cobalt:
  normal:
    launcher: qsub
    queue: normal
    options:
      - "-n 1"
      - "-t 10"

compilers:
  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran

```

If you want to retrieve full json schema file for buildtest configuration you can run `buildtest schema -n settings.schema.json --json` or short option `-j`.

3.9 Buildtest Schemas

3.9.1 CLI for buildtest schemas (`buildtest schema`)

buildtest uses JSON Schema for validating buildsspecs and *buildtest configuration file*. You can use `buildtest schema` command to see the list of schemas supported by buildtest. The schema files are denoted by `.schema.json` file extension.

```

$ buildtest schema
global.schema.json
definitions.schema.json
settings.schema.json
compiler-v1.0.schema.json
spack-v1.0.schema.json
script-v1.0.schema.json

```

Shown below is the command usage of `buildtest schema`

```

$ buildtest schema --help
usage: buildtest [options] [COMMANDS] schema [-h] [-e] [-j] [-n Schema Name]

```

(continues on next page)

(continued from previous page)

optional arguments:

```

-h, --help          show this help message and exit
-e, --example       Show schema examples
-j, --json          Display json schema file
-n Schema Name, --name Schema Name
                    show schema by name (e.g., script)

```

The json schemas are published at <https://buildtesters.github.io/buildtest/> and we provide a command line interface to view schema files and examples. You must use the `--name` option to select a schema, for instance if you want to view the JSON Schema for **script-v1.0.schema.json** you can run the following:

```
buildtest schema --name script-v1.0.schema.json --json
```

3.9.2 Schema Naming Convention

All schema files use the file extension **.schema.json** to distinguish itself as a json schema definition from an ordinary json file. The schema files are located in [buildtest/schemas](#) directory.

3.9.3 Schema Files

Definition Schema

This schema is used for declaring [definitions](#) that need to be reused in multiple schemas. We use `$ref` keyword to reference definitions from this file.

```

{
  "$id": "definitions.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "JSON Schema Definitions File. ",
  "description": "This file is used for declaring definitions that are referenced from_
↪ other schemas",
  "definitions": {
    "list_of_strings": {
      "type": "array",
      "uniqueItems": true,
      "minItems": 1,
      "items": {
        "type": "string"
      }
    },
    "string_or_list": {
      "oneOf": [
        {
          "type": "string"
        },
        {
          "$ref": "#/definitions/list_of_strings"
        }
      ]
    }
  }
}

```

(continues on next page)

(continued from previous page)

```

    "list_of_ints": {
        "type": "array",
        "uniqueItems": true,
        "minItems": 1,
        "items": {
            "type": "integer"
        }
    },
    "int_or_list": {
        "oneOf": [
            {
                "type": "integer"
            },
            {
                "$ref": "#/definitions/list_of_ints"
            }
        ]
    },
    "regex": {
        "type": "object",
        "description": "Perform regular expression search using ``re.search`` python_
↳ module on stdout/stderr stream for reporting if test ``PASS``. ",
        "required": [
            "stream",
            "exp"
        ],
        "additionalProperties": false,
        "properties": {
            "stream": {
                "type": "string",
                "enum": [
                    "stdout",
                    "stderr"
                ],
                "description": "The stream field can be stdout or stderr. buildtest will read_
↳ the output or error stream after completion of test and check if regex matches in_
↳ stream"
            },
            "exp": {
                "type": "string",
                "description": "Specify a regular expression to run with input stream_
↳ specified by ``stream`` field. buildtest uses re.search when performing regex"
            }
        }
    },
    "env": {
        "type": "object",
        "description": "One or more key value pairs for an environment (key=value)",
        "minItems": 1,
        "items": {
            "type": "object",
            "minItems": 1,

```

(continues on next page)

(continued from previous page)

```

    "propertyNames": {
      "pattern": "^[A-Za-z_][A-Za-z0-9_]*$"
    }
  },
  "description": {
    "type": "string",
    "description": "The ``description`` field is used to document what the test is_
↳ doing",
    "maxLength": 80
  },
  "tags": {
    "description": "Classify tests using a tag name, this can be used for categorizing_
↳ test and building tests using ``--tags`` option",
    "$ref": "#/definitions/string_or_list"
  },
  "skip": {
    "type": "boolean",
    "description": "The ``skip`` is a boolean field that can be used to skip tests_
↳ during builds. By default buildtest will build and run all tests in your builds_
↳ file, if ``skip: True`` is set it will skip the builds."
  },
  "executor": {
    "type": "string",
    "description": "Select one of the executor name defined in your configuration file_
↳ (``config.yml``). Every builds spec must have an executor which is responsible for_
↳ running job. "
  },
  "metrics_field": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "regex": {
        "$ref": "#/definitions/regex"
      },
      "vars": {
        "type": "string",
        "description": "Assign value to metric based on variable name"
      },
      "env": {
        "type": "string",
        "description": "Assign value to metric based on environment variable"
      }
    }
  },
  "metrics": {
    "type": "object",
    "description": "This field is used for defining one or more metrics that is_
↳ recorded for each test. A metric must have a unique name which is recorded in the test_
↳ metadata.",
    "patternProperties": {
      "^.*$": {

```

(continues on next page)

(continued from previous page)

```

        "$ref": "#/definitions/metrics_field",
        "description": "Name of metric"
    }
}
},
"run_only": {
    "type": "object",
    "description": "A set of conditions to specify when running tests. All conditions_
↳ must pass in order to process test.",
    "additionalProperties": false,
    "properties": {
        "scheduler": {
            "type": "string",
            "description": "Test will run only if scheduler is available. We assume_
↳ binaries are available in $PATH",
            "enum": [
                "lsf",
                "slurm",
                "cobalt",
                "pbs"
            ]
        },
        "user": {
            "type": "string",
            "description": "Test will run only if current user matches this field,_
↳ otherwise test will be skipped"
        },
        "platform": {
            "type": "string",
            "description": "This test will run if target system is Linux or Darwin. We_
↳ check target system using ``platform.system()`` and match with input field",
            "enum": [
                "Linux",
                "Darwin"
            ]
        },
        "linux_distro": {
            "type": "array",
            "description": "Specify a list of Linux Distros to check when processing test._
↳ If target system matches one of input field, test will be processed.",
            "uniqueItems": true,
            "minItems": 1,
            "items": {
                "type": "string",
                "enum": [
                    "darwin",
                    "ubuntu",
                    "debian",
                    "rhel",
                    "centos",
                    "fedora",
                    "sles",

```

(continues on next page)

(continued from previous page)

```

        "opensuse",
        "amazon",
        "arch"
    ]
}
}
},
"batch": {
    "type": "object",
    "description": "The ``batch`` field is used to specify scheduler agnostic_
↪ directives that are translated to #SBATCH or #BSUB based on your scheduler. This is an_
↪ experimental feature that supports a subset of scheduler parameters.",
    "additionalProperties": false,
    "properties": {
        "account": {
            "type": "string",
            "description": "Specify Account to charge job"
        },
        "begintime": {
            "type": "string",
            "description": "Specify begin time when job will start allocation"
        },
        "cpucount": {
            "type": "string",
            "description": "Specify number of CPU to allocate"
        },
        "email-address": {
            "type": "string",
            "description": "Email Address to notify on Job State Changes"
        },
        "exclusive": {
            "type": "boolean",
            "description": "Specify if job needs to run in exclusive mode"
        },
        "memory": {
            "type": "string",
            "description": "Specify Memory Size for Job"
        },
        "network": {
            "type": "string",
            "description": "Specify network resource requirement for job"
        },
        "nodecount": {
            "type": "string",
            "description": "Specify number of Nodes to allocate"
        },
        "qos": {
            "type": "string",
            "description": "Specify Quality of Service (QOS)"
        },
        "queue": {

```

(continues on next page)

(continued from previous page)

```

    "type": "string",
    "description": "Specify Job Queue"
  },
  "tasks-per-core": {
    "type": "string",
    "description": "Request number of tasks to be invoked on each core. "
  },
  "tasks-per-node": {
    "type": "string",
    "description": "Request number of tasks to be invoked on each node. "
  },
  "tasks-per-socket": {
    "type": "string",
    "description": "Request the maximum tasks to be invoked on each socket. "
  },
  "timelimit": {
    "type": "string",
    "description": "Specify Job timelimit"
  }
},
"status": {
  "type": "object",
  "description": "The status section describes how buildtest detects PASS/FAIL on
↳ test. By default returncode 0 is a PASS and anything else is a FAIL, however buildtest
↳ can support other types of PASS/FAIL conditions.",
  "additionalProperties": false,
  "properties": {
    "slurm_job_state": {
      "type": "string",
      "enum": [
        "COMPLETED",
        "FAILED",
        "OUT_OF_MEMORY",
        "TIMEOUT"
      ],
      "description": "This field can be used for checking Slurm Job State, if there
↳ is a match buildtest will report as ``PASS`` "
    },
    "returncode": {
      "description": "Specify a list of returncodes to match with script's exit code.
↳ buildtest will PASS test if script's exit code is found in list of returncodes. You
↳ must specify unique numbers as list and a minimum of 1 item in array",
      "$ref": "#/definitions/int_or_list"
    },
    "regex": {
      "$ref": "#/definitions/regex",
      "description": "Determine state (PASS/FAIL) of test based on regular
↳ expression on output or error stream"
    },
    "runtime": {
      "type": "object",

```

(continues on next page)

(continued from previous page)

```

        "description": "The runtime section will pass test based on min and max values.
↳and compare with actual runtime. ",
        "properties": {
            "min": {
                "type": "number",
                "minimum": 0,
                "description": "Specify a minimum runtime in seconds. The test will PASS.
↳if actual runtime exceeds min time."
            },
            "max": {
                "type": "number",
                "minimum": 0,
                "description": "Specify a maximum runtime in seconds. The test will PASS.
↳if actual runtime is less than max time"
            }
        }
    },
    "BB": {
        "$ref": "#/definitions/list_of_strings",
        "description": "Create burst buffer space, this specifies #BB options in your
↳test."
    },
    "DW": {
        "$ref": "#/definitions/list_of_strings",
        "description": "Specify Data Warp option (#DW) when using burst buffer."
    },
    "sbatch": {
        "$ref": "#/definitions/list_of_strings",
        "description": "This field is used for specifying #SBATCH options in test script."
    },
    "bsub": {
        "$ref": "#/definitions/list_of_strings",
        "description": "This field is used for specifying #BSUB options in test script."
    },
    "cobalt": {
        "$ref": "#/definitions/list_of_strings",
        "description": "This field is used for specifying #COBALT options in test script."
    },
    "pbs": {
        "$ref": "#/definitions/list_of_strings",
        "description": "This field is used for specifying #PBS directives in test script."
    },
    "executors": {
        "type": "object",
        "description": "Define executor specific configuration",
        "patternProperties": {
            "description": "Name of executor to override configuration",
            "^.*$": {
                "additionalProperties": false,
                "properties": {

```

(continues on next page)

(continued from previous page)

```

    "env": { "$ref": "#/definitions/env" },
    "vars": { "$ref": "#/definitions/env" },
    "sbatch": { "$ref": "#/definitions/list_of_strings" },
    "bsub": { "$ref": "#/definitions/list_of_strings" },
    "pbs": { "$ref": "#/definitions/list_of_strings" },
    "cobalt": { "$ref": "#/definitions/list_of_strings" },
    "BB": { "$ref": "#/definitions/BB" },
    "DW": { "$ref": "#/definitions/DW" },
    "status": { "$ref": "#/definitions/status" },
    "metrics": { "$ref": "#/definitions/metrics" }
  }
}
}
}
}
}

```

Settings Schema

This schema defines how *buildtest configuration* file is validated.

```

{
  "$id": "settings.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "buildtest configuration schema",
  "type": "object",
  "required": [
    "system"
  ],
  "additionalProperties": false,
  "properties": {
    "system": {
      "type": "object",
      "patternProperties": {
        "^.*$": {
          "$ref": "#/definitions/system"
        }
      }
    }
  },
  "definitions": {
    "system": {
      "required": [
        "executors",
        "moduletool",
        "load_default_buildspecs",
        "hostnames",
        "compilers"
      ],
      "additionalProperties": false,
      "type": "object",

```

(continues on next page)

(continued from previous page)

```

"properties": {
  "hostnames": {
    "type": "array",
    "description": "Specify a list of hostnames to check where buildtest can run.
↳for the given system record",
    "items": {
      "type": "string"
    }
  },
  "description": {
    "type": "string",
    "description": "system description field"
  },
  "buildspec_roots": {
    "type": "array",
    "items": {
      "type": "string"
    },
    "description": "Specify a list of directory paths to search buildspecs. This
↳field can be used with ``buildtest buildspec find`` to rebuild buildspec cache or
↳build tests using ``buildtest build`` command"
  },
  "load_default_buildspecs": {
    "type": "boolean",
    "description": "Specify whether buildtest should automatically load
↳buildspecs provided in buildtest repo into buildspec cache"
  },
  "testdir": {
    "type": "string",
    "description": "Specify full path to test directory where buildtest will write
↳tests."
  },
  "logdir": {
    "type": "string",
    "description": "Specify location where buildtest will write log files"
  },
  "moduletool": {
    "type": "string",
    "description": "Specify modules tool used for interacting with ``module``
↳command. ",
    "enum": [
      "environment-modules",
      "lmod",
      "N/A"
    ]
  },
  "processor": {
    "type": "object",
    "description": "Specify processor information",
    "additionalProperties": false,
    "properties": {
      "numcpus": {"type": "integer", "minimum": 1, "description": "Specify Total
↳Number of CPUs"}
    }
  }
}

```

(continues on next page)

(continued from previous page)

```

    "sockets": {"type": "integer", "minimum": 1, "description": "Specify Number_
↳ of CPU Sockets"},
    "cores": {"type": "integer", "minimum": 1, "description": "Specify Number of_
↳ Physical Cores"},
    "threads_per_core": {"type": "integer", "minimum": 1, "description":
↳ "Specify Threads per Core" },
    "core_per_socket": {"type": "integer", "minimum": 1, "description": "Specify_
↳ Cores per Socket"},
    "model": {"type": "string", "description": "Specify Processor Model" },
    "arch": {"type": "string", "description": "Specify processor architecture"},
    "vendor": {"type": "string", "description": "Vendor Name"}
  }
},
"compilers": {
  "type": "object",
  "description": "Declare compiler section for defining system compilers that_
↳ can be referenced in builds spec.",
  "additionalProperties": false,
  "properties": {
    "find": {
      "type": "object",
      "additionalProperties": false,
      "description": "Find compilers by specifying regular expression that is_
↳ applied to modulefile names",
      "properties": {
        "gcc": {
          "type": "string",
          "description": "Specify a regular expression to search for gcc_
↳ compilers from your module stack"
        },
        "intel": {
          "type": "string",
          "description": "Specify a regular expression to search for intel_
↳ compilers from your module stack"
        },
        "cray": {
          "type": "string",
          "description": "Specify a regular expression to search for cray_
↳ compilers from your module stack"
        },
        "clang": {
          "type": "string",
          "description": "Specify a regular expression to search for clang_
↳ compilers from your module stack"
        },
        "cuda": {
          "type": "string",
          "description": "Specify a regular expression to search for cuda_
↳ compilers from your module stack"
        },
        "pgi": {
          "type": "string",

```

(continues on next page)

(continued from previous page)

```

        "description": "Specify a regular expression to search for pgi_
↳compilers from your module stack"
    },
    "upcxx": {
        "type": "string",
        "description": "Specify a regular expression to search for upcxx_
↳compilers from your module stack"
    }
},
"compiler": {
    "type": "object",
    "additionalProperties": false,
    "description": "Start of compiler declaration",
    "properties": {
        "gcc": {
            "description": "Declaration of one or more GNU compilers where we_
↳define C, C++ and Fortran compiler. The GNU compiler wrapper are ``gcc``, ``g++`` and_
↳``gfortran``. ",
            "type": "object",
            "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↳section" } }
        },
        "intel": {
            "description": "Declaration of one or more Intel compilers where we_
↳define C, C++ and Fortran compiler. The Intel compiler wrapper are ``icc``, ``icpc``_
↳and ``ifort``. ",
            "type": "object",
            "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↳section" } }
        },
        "cray": {
            "description": "Declaration of one or more Cray compilers where we_
↳define C, C++ and Fortran compiler. The Cray compiler wrapper are ``cc``, ``CC`` and_
↳``ftn``.",
            "type": "object",
            "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↳section" } }
        },
        "pgi": {
            "description": "Declaration of one or more PGI compilers where we_
↳define C, C++ and Fortran compiler. The PGI compiler wrapper are ``pgcc``, ``pgc++``_
↳and ``pgfortran``.",
            "type": "object",
            "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↳section" } }
        },
        "clang": {
            "description": "Declaration of one or more Clang compilers where we_
↳define C, C++ compiler. The Clang compiler wrapper are ``clang``, ``clang++``.",
            "type": "object",
            "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↳section" } }
    }
}

```

(continues on next page)

(continued from previous page)

```

    },
    "cuda": {
        "description": "Declaration of one or more Cuda compilers where we
↪define C compiler. The Cuda compiler wrapper is ``nvcc``.",
        "type": "object",
        "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↪section" } }
    },
    "upcxx": {
        "description": "Declaration of one or more UPCXX compilers where we
↪define C, C++ compiler. The UPCXX compiler wrapper are ``upcxx``.",
        "type": "object",
        "patternProperties": { "^.*$": { "$ref": "#/definitions/compiler_
↪section" } }
    }
}
},
"executors": {
    "type": "object",
    "additionalProperties": false,
    "description": "The executor section is used for declaring your executors
↪that are responsible for running jobs. The executor section can be ``local``, ``lsf``,
↪``slurm``, ``cobalt``. The executors are referenced in builds spec using ``executor``
↪field.",
    "properties": {
        "defaults": {
            "type": "object",
            "description": "Specify default executor settings for all executors",
            "additionalProperties": false,
            "properties": {
                "pollinterval": {
                    "type": "integer",
                    "description": "Specify poll interval in seconds after job
↪submission, where buildtest will sleep and poll all jobs for job states. This field
↪can be configured based on your preference. Excessive polling every few seconds can
↪result in system degradation. ",
                    "minimum": 10,
                    "maximum": 300,
                    "default": 30
                },
                "launcher": {
                    "type": "string",
                    "enum": [
                        "sbatch",
                        "bsub",
                        "qsub"
                    ],
                    "description": "Specify batch launcher to use when submitting jobs,
↪this is applicable for LSF and Slurm Executors."
                }
            }
        }
    }
}

```

(continues on next page)

(continued from previous page)

```

        "max_pend_time": {
            "$ref": "#/definitions/max_pend_time"
        },
        "account": {
            "$ref": "#/definitions/account"
        },
        "max_jobs": {"$ref": "#/definitions/max_jobs"}
    },
    "local": {
        "type": "object",
        "description": "The ``local`` section is used for declaring local_
↳executors for running jobs on local machine",
        "patternProperties": {
            "^.*$": {
                "$ref": "#/definitions/local"
            }
        }
    },
    "lsf": {
        "type": "object",
        "description": "The ``lsf`` section is used for declaring LSF executors_
↳for running jobs using LSF scheduler",
        "patternProperties": {
            "^.*$": {
                "$ref": "#/definitions/lsf"
            }
        }
    },
    "slurm": {
        "type": "object",
        "description": "The ``slurm`` section is used for declaring Slurm_
↳executors for running jobs using Slurm scheduler",
        "patternProperties": {
            "^.*$": {
                "$ref": "#/definitions/slurm"
            }
        }
    },
    "cobalt": {
        "type": "object",
        "description": "The ``cobalt`` section is used for declaring Cobalt_
↳executors for running jobs using Cobalt scheduler",
        "patternProperties": {
            "^.*$": {
                "$ref": "#/definitions/cobalt"
            }
        }
    },
    "pbs": {
        "type": "object",
        "description": "The ``pbs`` section is used for declaring PBS executors_
↳for running jobs using PBS scheduler",

```

(continues on next page)

(continued from previous page)

```

        "patternProperties": {
            "^.*$": {
                "$ref": "#/definitions/pbs"
            }
        }
    },
    "cdash": {
        "type": "object",
        "description": "Specify CDASH configuration used to upload tests via
↪ 'buildtest cdash' command",
        "required": ["url", "project", "site"],
        "properties": {
            "url": {
                "type": "string",
                "description": "Url to CDASH server"
            },
            "project": {
                "type": "string",
                "description": "Name of CDASH project"
            },
            "site": {
                "type": "string",
                "description": "Site Name reported in CDASH"
            }
        }
    },
    "cc": {
        "description": "Specify path to C compiler wrapper. You may specify a compiler
↪ wrapper such as ``gcc`` assuming its in $PATH or you can use ``modules`` property to
↪ resolve path to compiler wrapper.",
        "type": "string"
    },
    "cxx": {
        "type": "string",
        "description": "Specify path to C++ compiler wrapper. You may specify a compiler
↪ wrapper such as ``g++`` assuming its in $PATH or you can use ``modules`` property to
↪ resolve path to compiler wrapper."
    },
    "fc": {
        "type": "string",
        "description": "Specify path to Fortran compiler wrapper. You may specify a
↪ compiler wrapper such as ``gfortran`` assuming its in $PATH or you can use ``modules``
↪ property to resolve path to compiler wrapper."
    },
    "max_jobs": {
        "description": "Maximum number of jobs that can be run at a given time for a
↪ particular executor",
        "type": "integer",

```

(continues on next page)

(continued from previous page)

```

    "minimum": 1
  },
  "compiler_section": {
    "description": "A compiler section is composed of ``cc``, ``cxx`` and ``fc``. ↵
↵ wrapper these are required when you need to specify compiler wrapper.",
    "type": "object",
    "additionalProperties": false,
    "required": [ "cc", "cxx", "fc" ],
    "properties": {
      "cc": { "$ref": "#/definitions/cc" },
      "cxx": { "$ref": "#/definitions/cxx" },
      "fc": { "$ref": "#/definitions/fc" },
      "module": { "$ref": "#/definitions/module" }
    }
  },
  "unique_string_array": {
    "type": "array",
    "uniqueItems": true,
    "items": {
      "type": "string"
    }
  },
  "disable": { "type": "boolean", "description": "Disable executor" },
  "module": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "purge": {
        "type": "boolean",
        "description": "Run ``module purge`` if purge is set"
      },
      "load": {
        "$ref": "definitions.schema.json#/definitions/list_of_strings",
        "description": "Load one or more modules via ``module load``"
      },
      "swap": {
        "description": "Swap modules using ``module swap``. The swap property expects ↵
↵ 2 unique modules.",
        "type": "array",
        "uniqueItems": true,
        "minItems": 2,
        "maxItems": 2,
        "items": { "type": "string" }
      }
    }
  },
  "script": {
    "type": "array",
    "additionalProperties": false,
    "items": { "type": "string" }
  },
  "max_pend_time": {

```

(continues on next page)

(continued from previous page)

```

    "type": "integer",
    "description": "Cancel job if it is still pending in queue beyond max_pend_time",
    "minimum": 10,
    "default": 90
  },
  "account": {
    "type": "string",
    "description": "Specify Job Account for charging resources"
  },
  "local": {
    "type": "object",
    "description": "An instance object of local executor",
    "additionalProperties": false,
    "required": [ "shell" ],
    "properties": {
      "description": {
        "type": "string",
        "description": "description field for documenting your executor"
      },
      "shell": {
        "type": "string",
        "description": "Specify the shell launcher you want to use when running tests.
↳ locally",
        "pattern": "^(/bin/bash|/bin/sh|/bin/csh|/bin/tcsh|/bin/
↳ zsh|sh|bash|csh|tcsh|zsh|python).*"
      },
      "before_script": { "$ref": "#/definitions/script" },
      "max_jobs": { "$ref": "#/definitions/max_jobs" },
      "disable": { "$ref": "#/definitions/disable" }
    }
  },
  "slurm": {
    "type": "object",
    "additionalProperties": false,
    "description": "An instance object of slurm executor",
    "properties": {
      "description": {
        "type": "string",
        "description": "description field for documenting your executor"
      },
      "launcher": {
        "type": "string",
        "enum": [ "sbatch" ],
        "description": "Specify the slurm batch scheduler to use. This overrides the
↳ default ``launcher`` field. This must be ``sbatch``. "
      },
      "options": {
        "type": "array",
        "items": { "type": "string" },
        "description": "Specify any other options for ``sbatch`` used by this executor.
↳ for running all jobs."
      }
    }
  },

```

(continues on next page)

(continued from previous page)

```

    "cluster": {
      "type": "string",
      "description": "Specify the slurm cluster you want to use ``-M <cluster>``"
    },
    "partition": {
      "type": "string",
      "description": "Specify the slurm partition you want to use ``-p <partition>``"
    },
    "qos": {
      "type": "string",
      "description": "Specify the slurm qos you want to use ``-q <qos>``"
    },
    "before_script": {
      "description": "The ``before_script`` section can be used to specify commands_
↳ before start of test. The script will be sourced in active shell.",
      "#ref": "#/definitions/script"
    },
    "max_pend_time": {
      "description": "overrides default ``max_pend_time`` value",
      "$ref": "#/definitions/max_pend_time"
    },
    "account": {
      "description": "overrides default ``account`` value",
      "$ref": "#/definitions/account"
    },
    "max_jobs": {"$ref": "#/definitions/max_jobs"},
    "disable": {"$ref": "#/definitions/disable"}
  },
  "lsf": {
    "type": "object",
    "description": "An instance object of lsf executor",
    "additionalProperties": false,
    "required": [ "queue" ],
    "properties": {
      "description": {
        "type": "string",
        "description": "description field for documenting your executor"
      },
      "launcher": {
        "type": "string",
        "enum": [ "bsub" ],
        "description": "Specify the lsf batch scheduler to use. This overrides the_
↳ default ``launcher`` field. It must be ``bsub``. "
      },
      "options": {
        "type": "array",
        "items": { "type": "string" },
        "description": "Specify any options for ``bsub`` for this executor when_
↳ running all jobs associated to this executor"
      },
      "queue": {

```

(continues on next page)

(continued from previous page)

```

        "type": "string",
        "description": "Specify the lsf queue you want to use ``-q <queue>``"
    },
    "before_script": {
        "description": "The ``before_script`` section can be used to specify commands_
↳ before start of test. The script will be sourced in active shell.",
        "#ref": "#/definitions/script"
    },
    "max_pend_time": {
        "description": "overrides default ``max_pend_time`` value",
        "$ref": "#/definitions/max_pend_time"
    },
    "account": {
        "description": "overrides default ``account`` value",
        "$ref": "#/definitions/account"
    },
    "max_jobs": {"$ref": "#/definitions/max_jobs"},
    "disable": {"$ref": "#/definitions/disable"}
}
},
"cobalt": {
    "type": "object",
    "description": "An instance object of cobalt executor",
    "additionalProperties": false,
    "required": [ "queue" ],
    "properties": {
        "description": {
            "type": "string",
            "description": "description field for documenting your executor"
        },
        "launcher": {
            "type": "string",
            "enum": [ "qsub" ],
            "description": "Specify the cobalt batch scheduler to use. This overrides the_
↳ default ``launcher`` field. It must be ``qsub``. "
        },
        "options": {
            "type": "array",
            "items": { "type": "string" },
            "description": "Specify any options for ``qsub`` for this executor when_
↳ running all jobs associated to this executor"
        },
        "queue": {
            "type": "string",
            "description": "Specify the lsf queue you want to use ``-q <queue>``"
        },
        "before_script": {
            "description": "The ``before_script`` section can be used to specify commands_
↳ before start of test. The script will be sourced in active shell.",
            "#ref": "#/definitions/script"
        },
        "max_pend_time": {

```

(continues on next page)

(continued from previous page)

```

        "description": "overrides default ``max_pend_time`` value",
        "$ref": "#/definitions/max_pend_time"
    },
    "account": {
        "description": "overrides default ``account`` value",
        "$ref": "#/definitions/account"
    },
    "max_jobs": {"$ref": "#/definitions/max_jobs"},
    "disable": {"$ref": "#/definitions/disable"}
}
},
"pbs": {
    "type": "object",
    "description": "An instance object of cobalt executor",
    "additionalProperties": false,
    "required": [ "queue" ],
    "properties": {
        "description": {
            "type": "string",
            "description": "description field for documenting your executor"
        },
        "launcher": {
            "type": "string",
            "enum": [ "qsub" ],
            "description": "Specify the pbs batch scheduler to use. This overrides the
↪ default ``launcher`` field. It must be ``qsub``. "
        },
        "options": {
            "type": "array",
            "items": { "type": "string" },
            "description": "Specify any options for ``qsub`` for this executor when
↪ running all jobs associated to this executor"
        },
        "queue": {
            "type": "string",
            "description": "Specify the lsf queue you want to use ``-q <queue>``"
        },
        "before_script": {
            "description": "The ``before_script`` section can be used to specify commands
↪ before start of test. The script will be sourced in active shell.",
            "$ref": "#/definitions/script"
        },
        "max_pend_time": {
            "description": "overrides default ``max_pend_time`` value",
            "$ref": "#/definitions/max_pend_time"
        },
        "account": {
            "description": "overrides default ``account`` value",
            "$ref": "#/definitions/account"
        },
        "max_jobs": {"$ref": "#/definitions/max_jobs"},
        "disable": {"$ref": "#/definitions/disable"}
    }
}

```

(continues on next page)

(continued from previous page)

```
}  
  }  
}  
}
```

Global Schema

This schema is used for validating buildspec file and validates outer level structure of test. This is referred as *Global Schema*

```
{  
  "$id": "global.schema.json",  
  "$schema": "http://json-schema.org/draft-07/schema#",  
  "title": "global schema",  
  "description": "buildtest global schema is validated for all buildspecs. The global_↵  
↵schema defines top-level structure of buildspec and defintions that are inherited for_↵  
↵sub-schemas",  
  "type": "object",  
  "required": ["version", "buildspecs"],  
  "additionalProperties": false,  
  "properties": {  
    "version": {  
      "type": "string",  
      "description": "The semver version of the schema to use (x.x)."  
    },  
    "maintainers": {  
      "type": "array",  
      "description": "One or more maintainers or aliases",  
      "uniqueItems": true,  
      "minItems": 1,  
      "items": {  
        "type": "string"  
      }  
    },  
    "buildspecs": {  
      "type": "object",  
      "description": "This section is used to define one or more tests (buildspecs). Each_↵  
↵test must be unique name",  
      "propertyNames": {  
        "pattern": "^[A-Za-z_.][A-Za-z0-9_.]*$",  
        "maxLength": 32  
      }  
    }  
  }  
}
```

Script Schema

This is the script schema used for writing scripts (bash, csh, sh, zsh, tcsh, python) and this is used for validating test instance when `type: script` is specified. For more details on script schema see [Script Schema](#).

```
{
  "$id": "script-v1.0.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "script schema version 1.0",
  "description": "The script schema is of ``type: script`` in sub-schema which is used
↳for running shell scripts",
  "type": "object",
  "required": ["type", "run", "executor"],
  "additionalProperties": false,
  "properties": {
    "type": {
      "type": "string",
      "pattern": "^script$",
      "description": "Select schema type to use when validating buildspec. This must be
↳of set to 'script'"
    },
    "description": { "$ref": "definitions.schema.json#/definitions/description" },
    "sbatch": { "$ref": "definitions.schema.json#/definitions/sbatch" },
    "bsub": { "$ref": "definitions.schema.json#/definitions/bsub" },
    "cobalt": { "$ref": "definitions.schema.json#/definitions/cobalt" },
    "pbs": { "$ref": "definitions.schema.json#/definitions/pbs" },
    "BB": { "$ref": "definitions.schema.json#/definitions/BB" },
    "DW": { "$ref": "definitions.schema.json#/definitions/DW" },
    "env": { "$ref": "definitions.schema.json#/definitions/env" },
    "vars": { "$ref": "definitions.schema.json#/definitions/env" },
    "executor": { "$ref": "definitions.schema.json#/definitions/executor" },
    "run_only": { "$ref": "definitions.schema.json#/definitions/run_only" },
    "shell": {
      "type": "string",
      "description": "Specify a shell launcher to use when running jobs. This sets the
↳shebang line in your test script. The ``shell`` key can be used with ``run`` section
↳to describe content of script and how its executed",
      "pattern": "^(/bin/bash|/bin/sh|/bin/csh|/bin/tcsh|/bin/
↳zsh|bash|sh|csh|tcsh|zsh|python).*"
    },
    "shebang": {
      "type": "string",
      "description": "Specify a custom shebang line. If not specified buildtest will
↳automatically add it in the test script."
    },
    "run": {
      "type": "string",
      "description": "A script to run using the default shell."
    },
    "status": { "$ref": "definitions.schema.json#/definitions/status" },
    "skip": { "$ref": "definitions.schema.json#/definitions/skip" },
    "tags": { "$ref": "definitions.schema.json#/definitions/tags" },
    "metrics": { "$ref": "definitions.schema.json#/definitions/metrics" },
  }
}
```

(continues on next page)

(continued from previous page)

```

    "executors": { "$ref": "definitions.schema.json#/definitions/executors" }
  }
}

```

Compiler Schema

This is the compiler schema used for validating buildsspecs that define test using `type: compiler`. This schema is used for compiling a single source code. For more details see [Compiler Schema](#)

```

{
  "$id": "compiler-v1.0.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "compiler schema version 1.0",
  "description": "The compiler schema is of ``type: compiler`` in sub-schema which is used for compiling and running programs",
  "type": "object",
  "required": [
    "type",
    "source",
    "compilers",
    "executor"
  ],
  "definitions": {
    "cc": {
      "type": "string",
      "description": "Set C compiler wrapper"
    },
    "fc": {
      "type": "string",
      "description": "Set Fortran compiler wrapper"
    },
    "cxx": {
      "type": "string",
      "description": "Set C++ compiler wrapper"
    },
    "cflags": {
      "type": "string",
      "description": "Set C compiler flags."
    },
    "fflags": {
      "type": "string",
      "description": "Set Fortran compiler flags."
    },
    "cxxflags": {
      "type": "string",
      "description": "Set C++ compiler flags."
    },
    "ldflags": {
      "type": "string",
      "description": "Set linker flags"
    }
  },
}

```

(continues on next page)

(continued from previous page)

```

"cppflags": {
  "type": "string",
  "description": "Set C or C++ preprocessor flags"
},
"pre_build": {
  "type": "string",
  "description": "Run commands before building program"
},
"post_build": {
  "type": "string",
  "description": "Run commands after building program"
},
"pre_run": {
  "type": "string",
  "description": "Run commands before running program"
},
"post_run": {
  "type": "string",
  "description": "Run commands after running program"
},
"run": {
  "type": "string",
  "description": "Run command for launching compiled binary"
},
"default_compiler_all": {
  "type": "object",
  "description": "Specify compiler configuration for all compiler groups. Use the
↳ ``all`` property if configuration is shared across compiler groups. This property can
↳ be overridden in compiler group or named compiler in ``config`` section.",
  "additionalProperties": false,
  "properties": {
    "sbatch": { "$ref": "definitions.schema.json#/definitions/sbatch" },
    "bsub": { "$ref": "definitions.schema.json#/definitions/bsub" },
    "cobalt": { "$ref": "definitions.schema.json#/definitions/cobalt" },
    "pbs": { "$ref": "definitions.schema.json#/definitions/pbs" },
    "BB": { "$ref": "definitions.schema.json#/definitions/BB" },
    "DW": { "$ref": "definitions.schema.json#/definitions/DW" },
    "env": { "$ref": "definitions.schema.json#/definitions/env" },
    "vars": { "$ref": "definitions.schema.json#/definitions/env" },
    "status": { "$ref": "definitions.schema.json#/definitions/status" },
    "pre_build": { "$ref": "#definitions/pre_build" },
    "post_build": { "$ref": "#definitions/post_build" },
    "pre_run": { "$ref": "#definitions/pre_run" },
    "post_run": { "$ref": "#definitions/post_run" },
    "run": { "$ref": "#definitions/run" }
  }
},
"default_compiler_config": {
  "type": "object",
  "description": "Specify compiler configuration for group of compilers. Use this
↳ property if you want to define common configuration for all compilers of same group.
↳ This property overrides ``all`` property. ",

```

(continues on next page)

(continued from previous page)

```

"properties": {
  "cc": { "$ref": "#definitions/cc" },
  "fc": { "$ref": "#definitions/fc" },
  "cxx": { "$ref": "#definitions/cxx" },
  "cflags": { "$ref": "#definitions/cflags" },
  "fflags": { "$ref": "#definitions/fflags" },
  "cxxflags": { "$ref": "#definitions/cxxflags" },
  "ldflags": { "$ref": "#definitions/ldflags" },
  "cppflags": { "$ref": "#definitions/cppflags" },
  "sbatch": { "$ref": "definitions.schema.json#/definitions/sbatch" },
  "bsub": { "$ref": "definitions.schema.json#/definitions/bsub" },
  "cobalt": { "$ref": "definitions.schema.json#/definitions/cobalt" },
  "pbs": { "$ref": "definitions.schema.json#/definitions/pbs" },
  "BB": { "$ref": "definitions.schema.json#/definitions/BB" },
  "DW": { "$ref": "definitions.schema.json#/definitions/DW" },
  "env": { "$ref": "definitions.schema.json#/definitions/env" },
  "vars": { "$ref": "definitions.schema.json#/definitions/env" },
  "status": { "$ref": "definitions.schema.json#/definitions/status" },
  "pre_build": { "$ref": "#definitions/pre_build" },
  "post_build": { "$ref": "#definitions/post_build" },
  "pre_run": { "$ref": "#definitions/pre_run" },
  "post_run": { "$ref": "#definitions/post_run" },
  "run": { "$ref": "#definitions/run" }
}
},
"compiler_declaration": {
  "type": "object",
  "description": "Specify compiler configuration at compiler level. The ``config``
↪ section has highest precedence when searching compiler configuration. This overrides
↪ fields found in compiler group and ``all`` property",
  "additionalProperties": false,
  "properties": {
    "cc": { "$ref": "#definitions/cc" },
    "fc": { "$ref": "#definitions/fc" },
    "cxx": { "$ref": "#definitions/cxx" },
    "cflags": { "$ref": "#definitions/cflags" },
    "fflags": { "$ref": "#definitions/fflags" },
    "cxxflags": { "$ref": "#definitions/cxxflags" },
    "ldflags": { "$ref": "#definitions/ldflags" },
    "cppflags": { "$ref": "#definitions/cppflags" },
    "sbatch": { "$ref": "definitions.schema.json#/definitions/sbatch" },
    "bsub": { "$ref": "definitions.schema.json#/definitions/bsub" },
    "cobalt": { "$ref": "definitions.schema.json#/definitions/cobalt" },
    "pbs": { "$ref": "definitions.schema.json#/definitions/pbs" },
    "BB": { "$ref": "definitions.schema.json#/definitions/BB" },
    "DW": { "$ref": "definitions.schema.json#/definitions/DW" },
    "env": { "$ref": "definitions.schema.json#/definitions/env" },
    "vars": { "$ref": "definitions.schema.json#/definitions/env" },
    "status": { "$ref": "definitions.schema.json#/definitions/status" },
    "pre_build": { "$ref": "#definitions/pre_build" },
    "post_build": { "$ref": "#definitions/post_build" },
    "pre_run": { "$ref": "#definitions/pre_run" },

```

(continues on next page)

(continued from previous page)

```

"post_run": { "$ref": "#definitions/post_run" },
"run": { "$ref": "#definitions/run" },

"module": {
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "purge": {
      "type": "boolean",
      "description": "Run ``module purge`` if purge is set"
    },
    "load": {
      "$ref": "definitions.schema.json#/definitions/list_of_strings",
      "description": "Load one or more modules via ``module load``"
    },
    "restore": {
      "description": "Load a collection name via ``module restore``",
      "type": "string"
    },
    "swap": {
      "description": "Swap modules using ``module swap``. The swap property
↳ expects 2 unique modules.",
      "type": "array",
      "uniqueItems": true,
      "minItems": 2,
      "maxItems": 2,
      "items": {
        "type": "string"
      }
    }
  }
},
"additionalProperties": false,
"properties": {
  "type": {
    "type": "string",
    "pattern": "^compiler$",
    "description": "Select schema type to use when validating buildspec. This must be
↳ of set to ``compiler``."
  },
  "description": {
    "$ref": "definitions.schema.json#/definitions/description"
  },
  "compilers": {
    "type": "object",
    "required": [
      "name"
    ],
    "additionalProperties": false,

```

(continues on next page)

(continued from previous page)

```

    "properties": {
      "name": {
        "description": "Specify a list of regular expression to search compiler_
↳instance from buildtest settings.",
        "$ref": "definitions.schema.json#/definitions/list_of_strings"
      },
      "exclude": {
        "description": "Specify a list of named compilers to exclude when building_
↳test based on regular expression specified in ``name`` property. The ``exclude``_
↳property has no effect if named compiler not found based on regular expression.",
        "$ref": "definitions.schema.json#/definitions/list_of_strings"
      },
      "default": {
        "type": "object",
        "additionalProperties": false,
        "properties": {
          "all": {
            "$ref": "#definitions/default_compiler_all"
          },
          "gcc": {
            "$ref": "#definitions/default_compiler_config"
          },
          "intel": {
            "$ref": "#definitions/default_compiler_config"
          },
          "pgi": {
            "$ref": "#definitions/default_compiler_config"
          },
          "cray": {
            "$ref": "#definitions/default_compiler_config"
          },
          "clang": {
            "$ref": "#definitions/default_compiler_config"
          },
          "cuda": {
            "$ref": "#definitions/default_compiler_config"
          },
          "upcxx": {
            "$ref": "#definitions/default_compiler_config"
          }
        }
      },
      "config": {
        "type": "object",
        "description": "Specify compiler configuration based on named compilers.",
        "patternProperties": {
          "^.*$": {
            "$ref": "#definitions/compiler_declaration"
          }
        }
      }
    }
  }
}

```

(continues on next page)

(continued from previous page)

```

    },
    "source": {
      "type": "string",
      "description": "Specify a source file for compilation, the file can be relative.
↳ path to buildspect or an absolute path"
    },
    "executor": {
      "$ref": "definitions.schema.json#/definitions/executor"
    },
    "run_only": {
      "$ref": "definitions.schema.json#/definitions/run_only"
    },
    "skip": {
      "$ref": "definitions.schema.json#/definitions/skip"
    },
    "tags": {
      "$ref": "definitions.schema.json#/definitions/tags"
    },
    "metrics": {
      "$ref": "definitions.schema.json#/definitions/metrics"
    }
  }
}

```

Spack Schema

This schema is used for writing tests with `spack` package manager using `type: spack` field. For more details see *Spack Schema*.

```

{
  "$id": "spack-v1.0.schema.json",
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "spack schema version 1.0",
  "description": "The spack schema is referenced using ``type: spack`` which is used for
↳ generating tests using spack package manager",
  "type": "object",
  "required": [
    "type",
    "executor",
    "spack"
  ],
  "additionalProperties": false,
  "properties": {
    "type": {
      "type": "string",
      "pattern": "^spack$",
      "description": "Select schema type to use when validating buildspect. This must be
↳ set to 'spack'"
    },
    "description": { "$ref": "definitions.schema.json#/definitions/description" },
    "executor": { "$ref": "definitions.schema.json#/definitions/executor" },
  }
}

```

(continues on next page)

(continued from previous page)

```

"env": { "$ref": "definitions.schema.json#/definitions/env" },
"vars": { "$ref": "definitions.schema.json#/definitions/env" },
"sbatch": { "$ref": "definitions.schema.json#/definitions/sbatch" },
"bsub": { "$ref": "definitions.schema.json#/definitions/bsub" },
"cobalt": { "$ref": "definitions.schema.json#/definitions/cobalt" },
"pbs": { "$ref": "definitions.schema.json#/definitions/pbs" },
"BB": { "$ref": "definitions.schema.json#/definitions/BB" },
"DW": { "$ref": "definitions.schema.json#/definitions/DW" },
"skip": { "$ref": "definitions.schema.json#/definitions/skip" },
"tags": { "$ref": "definitions.schema.json#/definitions/tags" },
"status": { "$ref": "definitions.schema.json#/definitions/status" },
"metrics": { "$ref": "definitions.schema.json#/definitions/metrics" },
"executors": { "$ref": "definitions.schema.json#/definitions/executors" },
"pre_cmds": {
  "type": "string",
  "description": "Shell commands run before spack"
},
"post_cmds": {
  "type": "string",
  "description": "Shell commands run after spack"
},

"spack": {
  "type": "object",
  "description": "Entry point to spack configuration",
  "required": [
    "root"
  ],
  "additionalProperties": false,
  "properties": {
    "root": {
      "type": "string"
    },
    "compiler_find": {
      "type": "boolean",
      "description": "Run ``spack compiler find`` if set to ``True``. This is run_
↳right after sourcing spack startup script."
    },
    "mirror": {
      "$ref": "definitions.schema.json#/definitions/env",
      "description": "Add mirror by running ``spack mirror add``"
    },
    "env": {
      "$ref": "#definitions/env",
      "description": "Manage spack environments via ``spack env`` command"
    },
    "install": {
      "$ref": "#definitions/install",
      "description": "Install spack packages by running ``spack install``. "
    },
    "verify_spack": {
      "type": "boolean",

```

(continues on next page)

(continued from previous page)

```

    "description": "This boolean will determine if we need to check for file_
↳existence where spack is cloned via ``root`` property and file **$SPACK_ROOT/share/
↳spack/setup-env.sh** exists. These checks can be disabled by setting this to ``False``_
↳which can be useful if you dont want buildtest to raise exception during test_
↳generation process and test is skipped.",
    "default": true
  },
  "test": {
    "$ref": "#definitions/test",
    "description": "Entry point to ``spack test``"
  }
}
},
"definitions": {
  "env": {
    "additionalProperties": false,
    "type": "object",
    "description": "Used for managing spack environment using ``spack env`` command. ",
    "properties": {
      "create": {
        "additionalProperties": false,
        "description": "Create a spack environment via ``spack env create``",
        "type": "object",
        "properties": {
          "remove_environment": {
            "type": "boolean",
            "description": "Remove existing spack environment before creating new_
↳environment. If set to ``True`` we will run ``spack env rm -y <name>``.",
            "default": false
          },
          "name": {
            "type": "string",
            "description": "Name of spack environment to create"
          },
          "manifest": {
            "type": "string",
            "description": "Specify path to spack manifest file (``spack.yaml`` or_
↳``spack.lock``) when creating environment"
          },
          "options": {
            "type": "string",
            "description": "Pass options to ``spack env create`` command"
          },
          "dir": {
            "type": "string",
            "description": "Create a spack environment in a specific directory. This_
↳will run ``spack env create -d <dir>``. Directory path does not have to exist prior to_
↳execution however user must have appropriate ACL in-order to create directory."
          }
        }
      }
    }
  },
},

```

(continues on next page)

(continued from previous page)

```

"activate": {
  "additionalProperties": false,
  "type": "object",
  "description": "Activate a spack environment via ``spack env activate``,
  "properties": {
    "name": {
      "type": "string",
      "description": "Name of spack environment to activate. In order to
↳ activate spack environment ``my-project`` you need to run ``spack env activate my-
↳ project`` which is specified by ``name: my-project``."
    },
    "options": {
      "type": "string",
      "description": "Pass options to ``spack env activate`` command"
    },
    "dir": {
      "type": "string",
      "description": "Activate spack environment from directory."
    }
  }
},
"rm": {
  "additionalProperties": false,
  "description": "Remove an existing spack environment via ``spack env rm``.",
  "type": "object",
  "required": [
    "name"
  ],
  "properties": {
    "name": {
      "type": "string",
      "description": "Remove spack environment by name. This will run ``spack
↳ env rm -y <name>``."
    }
  }
},
"mirror": {
  "$ref": "definitions.schema.json#/definitions/env",
  "description": "Add mirror in spack environment by running ``spack mirror add``
↳ "
},
"specs": {
  "$ref": "definitions.schema.json#/definitions/list_of_strings",
  "description": "Add specs to environment by running ``spack add <specs>``. The
↳ ``specs`` is a list of string which expect the argument to be name of spack package."
},
"concretize": {
  "type": "boolean",
  "description": "If ``concretize: true`` is set, we will concretize spack
↳ environment by running ``spack concretize -f`` otherwise this line will be ignored."
}
}

```

(continues on next page)

(continued from previous page)

```

    },
    "install": {
      "description": "Install spack packages using ``spack install`` command",
      "additionalProperties": false,
      "type": "object",
      "properties": {
        "options": {
          "type": "string",
          "description": "Pass options to ``spack install`` command"
        },
        "specs": {
          "$ref": "definitions.schema.json#/definitions/list_of_strings",
          "description": "List of specs to install using ``spack install`` command"
        }
      }
    },
    "test": {
      "type": "object",
      "additionalProperties": false,
      "required": [
        "run"
      ],
      "properties": {
        "remove_tests": {
          "type": "boolean",
          "description": "Remove all test suites in spack before running test via
↳ ``spack test run``. If set to ``True`` we will run ``spack test remove -y`` which will
↳ remove all test suites."
        },
        "run": {
          "description": "Run tests using spack via ``spack test run`` command. This
↳ command requires specs are installed in your spack instance prior to running tests.",
          "type": "object",
          "required": [
            "specs"
          ],
          "additionalProperties": false,
          "properties": {
            "option": {
              "type": "string",
              "description": "Pass options to ``spack test run``"
            },
            "specs": {
              "$ref": "definitions.schema.json#/definitions/list_of_strings",
              "description": "List of specs to run tests by running ``spack test run
↳ <specs>``."
            }
          }
        },
        "results": {
          "type": "object",
          "description": "View test results via ``spack test results`` after running
↳ tests via ``spack test run``. Results can be viewed using suite name or installed specs
↳ or both.",

```

(continues on next page)

(continued from previous page)

```

    "additionalProperties": false,
    "anyOf": [
      {"required": ["specs"] },
      {"required": ["suite"] },
      {"required": ["specs", "suite"] }
    ],
    "properties": {
      "option": {
        "type": "string",
        "description": "Pass options to ``spack test results``"
      },
      "suite": {
        "$ref": "definitions.schema.json#/definitions/list_of_strings",
        "description": "Report results by suite name by running ``spack test_
↳ results <suite>``."
      },
      "specs": {
        "$ref": "definitions.schema.json#/definitions/list_of_strings",
        "description": "Report result by spec name by running ``spack test run_
↳ -- <specs>``."
      }
    }
  }
}
}
}
}
}
}
}
}

```

3.9.4 Schema Examples

The schema examples are great way to help write your buildsspecs and help you understand the edge cases that can lead to an invalid buildspec. The schema examples are used in buildtest regression test for validating the schemas. We expose the examples through buildtest client so its accessible for everyone.

In order to view an example you can run:

```
buildtest schema -n <schema> --example
```

Settings Schema Examples

```

$ buildtest schema -n settings.schema.json --example
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/settings.schema.json/valid/slurm-example.yml
-----
system:
  generic:
    hostnames: ['.*']

    moduletool: lmod
    load_default_buildspecs: True

```

(continues on next page)

(continued from previous page)

```

buildspec_roots:
- $HOME/buildtest-cori
testdir: /tmp/buildtest
executors:
  defaults:
    pollinterval: 20
    launcher: sbatch
    max_pend_time: 30
    account: admin
  slurm:
    normal:
      options: ["-C haswell"]
      qos: normal
      before_script: |
        time
        echo "commands run before job"

compilers:
  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/settings.schema.json/valid/cobalt-example.yml

```

system:
  generic:
    hostnames: ['.*']

  moduletool: lmod
  load_default_buildspecs: True
  executors:
    defaults:
      launcher: qsub
      max_pend_time: 30

  cobalt:
    knl:
      queue: knl

    haswell:
      queue: haswell

  compilers:
    compiler:
      gcc:
        default:
          cc: /usr/bin/gcc
          cxx: /usr/bin/g++
          fc: /usr/bin/gfortran

```

(continues on next page)

(continued from previous page)

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/settings.schema.json/valid/pbs-example.yml

```
system:
  generic:
    hostnames: ['.*']

    moduletool: N/A
    load_default_buildspecs: True
    executors:
      defaults:
        pollinterval: 10
        launcher: qsub
        max_pend_time: 30
      pbs:
        workq:
          queue: workq
    compilers:
      compiler:
        gcc:
          default:
            cc: /usr/bin/gcc
            cxx: /usr/bin/g++
            fc: /usr/bin/gfortran
```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/settings.schema.json/valid/lsf-example.yml

```
system:
  generic:
    hostnames: ['.*']

    moduletool: lmod
    load_default_buildspecs: False
    executors:
      defaults:
        pollinterval: 10
        launcher: bsub
        max_pend_time: 45
      lsf:
        batch:
          description: "LSF Executor name 'batch' that submits jobs to 'batch' queue"
          queue: batch
          account: developer
          options: ["-W 20"]
          before_script: |
            time
            echo "commands run before job"
        test:
          description: "LSF Executor name 'test' that submits jobs to 'test' queue"
          launcher: bsub
          queue: test
          account: qa
```

(continues on next page)

(continued from previous page)

```

    options: ["-W 20"]
compilers:
  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/settings.schema.json/valid/local-executor.yml

```

```

system:
  generic:
    hostnames: ['.*']

    logdir: $BUILDTEST_ROOT/logs
    testdir: $BUILDTEST_ROOT/tests

    moduletool: N/A
    load_default_buildspecs: False
    cdash:
      url: https://my.cdash.org
      project: buildtest
      site: laptop
    processor:
      numcpus: 8
      cores: 4
      threads_per_core: 2
      sockets: 1
      model: "Intel(R) Core(TM) i7-8569U CPU @ 2.80GHz"
    executors:
      local:
        bash:
          description: submit jobs on local machine using bash shell
          shell: bash
          before_script: |
            time
            echo "commands run before job"

        sh:
          description: submit jobs on local machine using sh shell
          shell: sh

        csh:
          description: submit jobs on local machine using csh shell
          shell: csh -x

        tcsh:
          description: submit jobs on local machine using tcsh shell
          shell: /bin/tcsh

        zsh:

```

(continues on next page)

(continued from previous page)

```

description: submit jobs on local machine using zsh shell
shell: /bin/zsh

python:
description: submit jobs on local machine using python shell
shell: python

compilers:
find:
gcc: "(gcc|GCC|PrgEnv-gnu)"
intel: "(intel|Intel|PrgEnv-intel)"
cray: "(cray|PrgEnv-cray)"
clang: "(clang|Clang)"
cuda: "(cuda|CUDA)"
pgi: "(pgi|PGI|PrgEnv-pgi)"

compiler:
gcc:
default:
cc: /usr/bin/gcc
cxx: /usr/bin/g++
fc: /usr/bin/gfortran
gcc@7.2.0:
cc: 'cc'
cxx: 'c++'
fc: 'fc'
module:
load:
- gcc/7.2.0
intel:
intel@2019:
cc: 'icc'
cxx: 'icpc'
fc: 'ifort'
module:
purge: True
load:
- gcc/7.2.0
- intel/2019
cray:
craype@2.6.2:
cc: 'cc'
cxx: 'CC'
fc: 'fc'
module:
load: [craype/2.6.2]
swap: [PrgEnv-gnu, PrgEnv-cray]

clang:
clang@12.0.0:
cc: 'clang'
cxx: 'clang++'

```

(continues on next page)

(continued from previous page)

```

    fc: 'None'
    module:
      load: [clang/12.0]
  cuda:
    cuda@11.0:
      cc: 'nvcc'
      cxx: 'nvcc'
      fc: 'None'
      module:
        load: [cuda/11.0]
  pgi:
    pgi@18.0:
      cc: 'pgcc'
      cxx: 'pgc++'
      fc: 'pgfortran'
      module:
        load: [pgi/18.0]

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/settings.schema.json/valid/combined_executor.yml

```

system:
  generic:
    hostnames: ['.*']

  moduletool: N/A
  load_default_buildspecs: True
  executors:
    local:
      bash:
        description: submit jobs on local machine
        shell: bash -v

    slurm:
      haswell:
        launcher: sbatch
        options:
          - "-p haswell"
          - "-t 00:10"

    lsf:
      batch:
        launcher: bsub
        queue: batch
        options:
          - "-q batch"
          - "-t 00:10"

    cobalt:
      normal:
        launcher: qsub
        queue: normal
        options:

```

(continues on next page)

(continued from previous page)

```
- "-n 1"
- "-t 10"

compilers:
  compiler:
    gcc:
      default:
        cc: /usr/bin/gcc
        cxx: /usr/bin/g++
        fc: /usr/bin/gfortran
```

Global Schema Examples

```
$ buildtest schema -n global.schema.json --example
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/global.schema.json/valid/examples.yml
```

```
version: "1.0"
```

```
buildspecs:
```

```
  # testing all caps
```

```
  ABCDEFGHIJKLMNOPQRSTUVWXYZ:
```

```
    type: script
```

```
    run: "hostname"
```

```
  # testing all lowercase letters
```

```
  abcdefghijklmnopqrstuvwxyz:
```

```
    type: script
```

```
    run: "hostname"
```

```
  # testing '_' in beginning followed by all numbers
```

```
  _0123456789:
```

```
    type: script
```

```
    run: "hostname"
```

```
  # testing '_' in combination with caps, lowercase and numbers
```

```
  _ABCDEFabcdef0123456789:
```

```
    type: script
```

```
    run: "hostname"
```

```
  # testing '_' at end of word
```

```
  abcdefghijklmnopqrstuvwxyz_:
```

```
    type: script
```

```
    run: "hostname"
```

```
  # testing '.' in beginning of word
```

```
  .helloworld:
```

```
    type: script
```

```
    run: hostname
```

(continues on next page)

(continued from previous page)

```
# testing '.' in middle of word
hello.world:
  type: script
  run: hostname
```

```
# testing '.' at end of word
helloworld.:
  type: script
  run: hostname
```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/global.schema.json/invalid/missing-version.yml

```
buildspecs:
```

```
# Shell would be accepted to indicate a single line shell command (or similar)
login_node_check:
  type: script
  run: "ping login 1"
```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/global.schema.json/invalid/exceed_testname_length.yml

```
# this test fails because it exceeds 32 character length for test name
version: "1.0"
```

```
buildspecs:
  this_test_exceeds_32_character_length:
    type: script
    run: hostname
```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/global.schema.json/invalid/unique_maintainers.yml

```
version: "1.0"
maintainers: [shahzebsiddiqui, shahzebsiddiqui]
buildspecs:
  hostname:
    type: script
    run: "hostname"
```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/global.schema.json/invalid/maintainers_type_mismatch.yml

```
version: "1.0"
# wrong type for maintainers key, expects a string
maintainers: 1
buildspecs:
  hostname:
    type: script
    run: "hostname"
```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪buildtest/schemas/examples/global.schema.json/invalid/invalid_pattern.yml

```
version: "1.0"
buildspecs:
  # invalid pattern for test. Must be matching regex "[A-Za-z_][A-Za-z0-9_]*$" when
  ↪declaring a dict
```

(continues on next page)

(continued from previous page)

```
(badname:
  type: script
  run: "ping login 1"
```

Script Schema Examples

```
$ buildtest schema -n script-v1.0.schema.json --example
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/script-v1.0.schema.json/valid/examples.yml
```

```
-----
version: "1.0"
buildspecs:
  multiline_run:
    executor: generic.local.bash
    type: script
    description: multiline run command
    run: |
      echo "1"
      echo "2"

  single_command_run:
    executor: generic.local.bash
    type: script
    description: single command as a string for run command
    run: "hostname"

  declare_env:
    executor: generic.local.bash
    type: script
    description: declaring environment variables
    env:
      FOO: BAR
      X: 1
    run: |
      echo $FOO
      echo $X

  declare_vars:
    executor: generic.local.bash
    type: script
    description: declaring variables
    vars:
      First: Bob
      Last:  Bill
    run: |
      echo "First:" $First
      echo "Last:" $Last

  declare_shell_sh:
```

(continues on next page)

(continued from previous page)

```
executor: generic.local.sh
type: script
description: declare shell name to sh
shell: sh
run: hostname

declare_shell_bash:
  executor: generic.local.bash
  type: script
  description: declare shell name to bash
  shell: bash
  run: hostname

declare_shell_python:
  executor: generic.local.python
  type: script
  description: declare shell name to python
  shell: python
  run: |
    print("Hello World")

declare_shell_bin_bash:
  executor: generic.local.bash
  type: script
  description: declare shell name to /bin/bash
  shell: "/bin/bash -e"
  run: hostname

declare_shell_name_bin_sh:
  executor: generic.local.sh
  type: script
  description: declare shell name to /bin/sh
  shell: "/bin/sh -e"
  run: hostname

declare_shell_opts:
  executor: generic.local.sh
  type: script
  description: declare shell name to sh
  shell: "sh -e"
  run: hostname

declare_shell_bin_zsh:
  executor: generic.local.zsh
  type: script
  description: declare shell zsh
  shell: "zsh"
  run: hostname

declare_shell_zsh:
  executor: generic.local.zsh
  type: script
```

(continues on next page)

(continued from previous page)

```
description: declare shell /bin/zsh
shell: "zsh"
run: hostname
```

```
declare_shell_bin_csh:
  executor: generic.local.csh
  type: script
  description: declare shell /bin/csh
  shell: "/bin/csh"
  run: hostname
```

```
declare_shell_csh:
  executor: generic.local.csh
  type: script
  description: declare shell /bin/tcsh
  shell: "csh"
  run: hostname
```

```
declare_shell_bin_tcsh:
  executor: generic.local.csh
  type: script
  description: declare shell /bin/tcsh
  shell: "/bin/tcsh"
  run: hostname
```

```
declare_shell_tcsh:
  executor: generic.local.csh
  type: script
  description: declare shell tcsh
  shell: "tcsh"
  run: hostname
```

```
declare_shebang:
  executor: generic.local.bash
  type: script
  description: declare shell name to sh
  shebang: "#!/usr/bin/env bash"
  run: hostname
```

```
status_returncode_list:
  executor: generic.local.bash
  type: script
  description: The returncode can be a list of integers
  run: exit 0
  status:
    returncode: [0]
```

```
status_returncode_int:
  executor: generic.local.bash
  type: script
  description: The returncode can be an integer to match with single returncode
```

(continues on next page)

(continued from previous page)

```

run: exit 0
status:
  returncode: 0

status_regex:
  executor: generic.local.bash
  type: script
  description: This test pass with a regular expression status check
  run: hostname
  status:
    regex:
      stream: stdout
      exp: "^$"

status_regex_returncode:
  executor: generic.local.bash
  type: script
  description: This test fails because returncode and regex specified
  run: hostname
  status:
    returncode: [0]
    regex:
      stream: stdout
      exp: "^hello"

status_runtime_min_max:
  type: script
  executor: generic.local.sh
  description: "Run a sleep job for 2 seconds and test pass if its within 1.0-4.0sec"
  tags: ["tutorials"]
  run: sleep 2
  status:
    runtime:
      min: 1.0
      max: 4.0

status_runtime_min:
  type: script
  executor: generic.local.sh
  description: "Run a sleep job for 2 seconds and test pass if exceeds mintime of 1.
↪0sec"
  tags: ["tutorials"]
  run: sleep 2
  status:
    runtime:
      min: 1.0

status_runtime_max:
  type: script
  executor: generic.local.sh
  description: "Run a sleep job for 2 seconds and test pass if less than maxtime of 4.
↪0sec"

```

(continues on next page)

(continued from previous page)

```
tags: ["tutorials"]
run: sleep 2
status:
  runtime:
    max: 4.0

sbatch_example:
  type: script
  executor: generic.local.bash
  description: This test runs hostname using sbatch directives
  sbatch:
    - "-t 10:00:00"
    - "-p normal"
    - "-N 1"
    - "-n 8"
  run: hostname

bsub_example:
  type: script
  executor: generic.local.bash
  description: This test runs hostname using bsub directives
  bsub:
    - "-W 00:30"
    - "-N 1"
  run: hostname

cobalt_example:
  type: script
  executor: generic.local.bash
  description: This test runs hostname using cobalt directives
  cobalt:
    - "-t 30"
    - "-n 1"
  run: hostname

skip_example:
  type: script
  executor: generic.local.bash
  description: this test is skip
  skip: true
  run: hostname

tag_str_example:
  type: script
  executor: generic.local.bash
  description: tags can be defined as string
  tags: network
  run: hostname

tag_list_example:
  type: script
  executor: generic.local.bash
```

(continues on next page)

(continued from previous page)

```

description: This is a tag example using list
sbatch:
  - "-t 10:00:00"
  - "-p normal"
  - "-N 1"
  - "-n 8"
tags: ["slurm"]
run: hostname

run_only_example:
  type: script
  executor: generic.local.bash
  description: run_only example that runs with user1 on Linux system (rhel, centos)
↪with LSF
  run_only:
    user: user1
    scheduler: lsf
    platform: Linux
    linux_distro:
      - rhel
      - centos
  run: |
    uname -av
    lsinfo

metrics_regex_example:
  type: script
  executor: generic.local.bash
  description: metrics regular expression example
  run: echo "HPCG result is VALID with a GFLOP/s rating of=63.6515"
  metrics:
    hpcg_rating:
      regex:
        exp: 'rating of=(\d+\.\d+)$'
        stream: stdout

metric_variable_assignment:
  executor: generic.local.sh
  type: script
  description: capture result metric based on variables and environment variable
  vars:
    GFLOPS: "63.6515"
  env:
    FOO: BAR
  run: |
    echo $GFLOPS
    echo $FOO
  tags: tutorials
  metrics:
    gflops:
      vars: "GFLOPS"
  foo:

```

(continues on next page)

(continued from previous page)

```

    env: "FOO"

multi_executor_vars:
  type: script
  executor: 'generic.local.(sh|bash)'
  description: single test multiple executor with variable declaration
  run: |
    echo $X
    echo $Y
  executors:
    generic.local.sh:
      vars:
        X: 1
        Y: 2
    generic.local.bash:
      vars:
        X: 10
        Y: 11

multi_executor_environment:
  type: script
  executor: 'generic.local.(sh|bash)'
  description: single test multiple executor with environment declaration
  run: echo $SHELL
  executors:
    generic.local.sh:
      env:
        SHELL: sh
    generic.local.bash:
      env:
        SHELL: bash

executors_sbbatch_declaration:
  type: script
  executor: 'generic.local.(bash|sh)'
  description: Declaring sbatch by executors section
  tags: [tutorials]
  run: hostname
  sbatch: ["-N 4"]
  executors:
    generic.local.bash:
      sbatch: ["-n 4", "-N 1", "-t 30"]
    generic.local.sh:
      sbatch: ["-n 8", "-N 1", "-t 60"]

executors_bsub_declaration:
  type: script
  executor: 'generic.local.(bash|sh)'
  description: Declaring bsub by executors section
  tags: [tutorials]
  run: hostname
  executors:

```

(continues on next page)

(continued from previous page)

```

generic.local.bash:
  bsub: ["-n 4", "-W 30"]
generic.local.sh:
  bsub: ["-n 8", "-W 60"]

executors_pbs_declaration:
  type: script
  executor: 'generic.local.(bash|sh)'
  description: Declaring pbs by executors section
  tags: [tutorials]
  run: hostname
  executors:
    generic.local.bash:
      pbs: ["-l ncpus=4", "-l walltime=30"]
    generic.local.sh:
      pbs: ["-l ncpus=8", "-l walltime=60"]

executors_status_declaration:
  type: script
  executor: 'generic.local.(bash|sh)'
  description: Declaring status by executors section
  tags: [tutorials]
  run: exit 0
  executors:
    generic.local.bash:
      status:
        returncode: 0
    generic.local.sh:
      status:
        returncode: [1, 2]

executors_metrics_declaration:
  type: script
  executor: 'generic.local.(bash|sh)'
  description: Declaring metrics by executors section
  tags: [tutorials]
  run: echo "Hello World"
  executors:
    generic.local.bash:
      metrics:
        hello:
          regex:
            stream: stdout
            exp: "(Hello)"
    generic.local.sh:
      metrics:
        world:
          regex:
            stream: stdout
            exp: "(World)"

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/script-v1.0.schema.json/invalid/examples.yml

(continues on next page)

(continued from previous page)

```
-----
version: "1.0"
buildspecs:
  invalid_test_name_&!@#%$:
    type: script
    executor: generic.local.bash
    description: "invalid test name"

  invalid_bash:
    type: script
    executor: generic.local.bash
    shell: "bash-missing-run"

  missing_run_key:
    type: script
    executor: generic.local.bash
    description: invalid key name roon, missing run key
    roon: |
      systemctl is-active slurmd
      systemctl is-enabled slurmd | grep enabled

  invalid_env_type:
    type: script
    executor: generic.local.bash
    description: env key should be a dictionary
    env:
      - FOO=BAR
    run: echo $FOO

  invalid_vars_type:
    type: script
    executor: generic.local.bash
    description: var key should be a dictionary
    vars:
      - FOO=BAR
    run: echo $FOO

  invalid_description:
    type: script
    executor: generic.local.bash
    description:
      - "Multi Line description"
      - "is not accepted"

  invalid_regex_stream:
    type: script
    executor: generic.local.bash
    description: This test fails because of invalid regex stream
    run: hostname
    status:
      regex:
```

(continues on next page)

(continued from previous page)

```
    stream: file
    exp: "world$"

regex_additionalProperties_test:
  type: script
  executor: generic.local.bash
  description: Testing for additional properties in regex field
  run: hostname
  status:
    regex:
      stream: stdout
      exp: "world$"
      X: 1

missing_regex_exp:
  type: script
  executor: generic.local.bash
  description: This test fails because of missing key 'exp' in regex
  run: hostname
  status:
    regex:
      stream: stdout

invalid_returncode_type:
  type: script
  executor: generic.local.bash
  description: This test fails because of invalid return code type
  run: hostname
  status:
    returncode: ["1"]

empty_returncode_list:
  type: script
  executor: generic.local.bash
  description: An empty returncode list will cause an error
  run: hostname
  status:
    returncode: []

non_int_returncodes:
  type: script
  executor: generic.local.bash
  description: The returncode must be an int and not a number
  run: exit 1
  status:
    returncode: 1.01

non_int_returncodes_list:
  type: script
  executor: generic.local.bash
  description: The returncode must be a list of integers and no numbers
```

(continues on next page)

(continued from previous page)

```
run: exit 1
status:
  returncode: [1, 2.230]

invalid_shell_usr_bin_bash:
  type: script
  executor: generic.local.bash
  description: invalid shell name '/usr/bin/bash'
  shell: /usr/bin/bash
  run: hostname

invalid_shell_type:
  type: script
  executor: generic.local.bash
  description: invalid shell type must be a string
  shell: ["/bin/bash"]
  run: hostname

invalid_type_shell_shebang:
  type: script
  executor: generic.local.bash
  description: invalid type for shell shebang, must be a string
  shebang: ["#!/bin/bash"]
  run: hostname

invalid_skip_value:
  type: script
  executor: generic.local.bash
  description: invalid value for skip, must be boolean
  skip: 1
  run: hostname

empty_tags:
  type: script
  executor: generic.local.bash
  description: tag list can't be empty, requires one item.
  tags: []
  run: hostname

non_unique_tags:
  type: script
  executor: local.bash
  description: tag names must be unique
  tags: ["network", "network"]
  run: hostname

invalid_tags_value:
  type: script
  executor: generic.local.bash
  description: invalid tag value must be all string items
  tags: ["network", 400 ]
  run: hostname
```

(continues on next page)

(continued from previous page)

```

additionalProperties_test:
  type: script
  executor: generic.local.bash
  description: additional properties are not allowed so any invalid key/value pair_
↳will result in error
  FOO: BAR
  run: hostname

additionalProperties_status:
  type: script
  executor: generic.local.bash
  description: test additional properties in status object. This is not allowed
  sbatch: [ "-n 2", "-q normal", "-t 10"]
  run: hostname
  status:
    slurm_job_state: "COMPLETED"
    FOO: BAR

invalid_runtime_min:
  type: script
  executor: generic.local.sh
  description: "Invalid type for min property in runtime. Must be an integer or float_
↳not a string"
  run: sleep 2
  status:
    runtime:
      min: "1"

runtime_min_must_exceed_0:
  type: script
  executor: generic.local.sh
  description: "The runtime must exceed 0"
  run: sleep 2
  status:
    runtime:
      min: -1

invalid_slurm_job_state:
  type: script
  executor: generic.local.sh
  description: invalid value for slurm_job_state, should raise error with enum values.
  sbatch:
    - "-n 2"
    - "-q normal"
    - "-t 10"
  run: hostname
  status:
    slurm_job_state: "FINISH"

duplicate_linux_distro:
  type: script

```

(continues on next page)

(continued from previous page)

```

executor: generic.local.bash
description: Duplicate items in linux_distro is not allowed
run_only:
  linux_distro:
    - rhel
    - rhel
run: uname -av

empty_list_linux_distro:
  type: script
  executor: generic.local.bash
  description: Empty List in linux_distro is not allowed. Requires atleast 1 item
  run_only:
    linux_distro: []
  run: uname -av

additionalProperties_run_only:
  type: script
  executor: generic.local.bash
  description: additional Properties not allowed in run_only field. Invalid field.
↪python
  run_only:
    user: root
    python: 3.5
  run: hostname

invalid_metrics_additional_property:
  type: script
  executor: generic.local.bash
  description: Test for additional property for metrics property
  vars:
    FOO: BAR
  run: echo $FOO
  metrics:
    foo:
      variable: FOO

invalid_metrics_type:
  type: script
  executor: generic.local.bash
  description: metrics property is an object, testing for type
  vars:
    FOO: BAR
  run: echo $FOO
  metrics: FOO

executors_invalid_var_type:
  type: script
  executor: "generic.local.(bash|sh|zsh)"
  description: Invalid type field for 'vars'
  tags: [tutorials]
  run: echo $FOO

```

(continues on next page)

(continued from previous page)

```

executors:
  generic.local.bash:
    vars: ["FOO=BAR"]

executors_additionalProperties:
  type: script
  executor: "generic.local.(bash|sh|zsh)"
  description: Testing for additional properties in 'executors'
  tags: [tutorials]
  run: hostname
  sbatch: ["-N 4"]
  executors:
    generic.local.bash:
      sbatch: ["-n 4", "-N 1", "-t 30"]
      FOO: BAR
    generic.local.sh:
      sbatch: ["-n 8", "-N 1", "-t 60"]
    generic.local.zsh:
      sbatch: ["-n 16", "-N 2", "-t 120"]

```

Compiler Schema Examples

```

$ buildtest schema -n compiler-v1.0.schema.json --example
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/compiler-v1.0.schema.json/valid/examples.yml

```

```

-----
version: "1.0"
buildspecs:
  gnu_example:
    executor: local.bash
    type: compiler
    description: "gnu example with modules, and cflags example"
    source: src/hello.c
    compilers:
      name: [gcc]
      config:
        gcc@8.4.0:
          cflags: "-O3"

  intel_example:
    executor: local.bash
    type: compiler
    description: "intel example using cflags"
    source: src/hello.c
    compilers:
      name: [intel]
      config:
        intel@2018:
          cflags: "-O1"

```

(continues on next page)

(continued from previous page)

```

clang_example:
  executor: local.bash
  type: compiler
  description: "clang example using cflags"
  source: src/hello.c
  compilers:
    name: [clang]
    default:
      clang:
        cflags: "-O1"
    config:
      clang@11:
        cflags: "-O2"

upcxx_example:
  executor: local.bash
  type: compiler
  description: "upcxx compiler declaration in default and config section "
  source: src/hello.c
  compilers:
    name: [upcxx]
    default:
      upcxx:
        cflags: "-g aries"
    config:
      upcxx@2020:
        cflags: "-O1 -g aries"

pgi_example:
  executor: local.bash
  type: compiler
  description: "pgi example using cxxflags, ldflags in default and config section"
  source: src/hello.cpp
  compilers:
    name: ["^(pgi|PrgEnv)"]
    default:
      pgi:
        cxxflags: "-O1"
        ldflags: "-lm"
    config:
      pgi@18.1:
        module:
          swap: [PrgEnv-gnu, PrgEnv-pgi]
          load: [pgi/18.1]
      pgi@18.2:
        module:
          swap: [PrgEnv-gnu, PrgEnv-pgi]
          load: [pgi/18.2]

cray_example:
  executor: local.bash
  type: compiler

```

(continues on next page)

(continued from previous page)

```

description: "cray example using fflags and cppflags"
source: src/hello.f90
compilers:
  name: ["PrgEnv-cray"]
  default:
    cray:
      fflags: "-O1"
  config:
    PrgEnv-cray@2.6.2:
      module:
        swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

sbatch_example_all_compiler_groups:
  type: compiler
  description: sbatch example to for all compiler groups
  executor: local.bash
  source: src/hello.f90
  compilers:
    name: ["PrgEnv-cray"]
    default:
      cray:
        fflags: "-O1"
      all:
        sbatch: ["-t 10", "-n 2", "-C haswell" ]
  config:
    PrgEnv-cray@2.6.2:
      module:
        swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

bsub_all_compiler_groups:
  type: compiler
  description: bsub example for all compiler groups
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: [intel]
    default:
      all:
        bsub: ["-W 00:30", "-n 2"]
  config:
    intel@2019:
      cxxflags: "-O1"

cobalt_all_compiler_groups:
  type: compiler
  description: cobalt example for all compiler groups
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: [intel]
    default:

```

(continues on next page)

(continued from previous page)

```

    all:
      cobalt: ["-t 30", "-n 1"]
    config:
      intel@2019:
        cxxflags: "-O1"

sbatch_compiler_group:
  type: compiler
  description: sbatch example in multiple compiler groups.
  executor: local.bash
  source: src/hello.f90
  compilers:
    name: ["^(gcc|intel)"]
    default:
      gcc:
        fflags: "-O1"
        sbatch: ["-t 10", "-n 2", "-C haswell" ]
      intel:
        fflags: "-O2"
        sbatch: ["-t 10", "-n 2", "-C knl" ]
    config:
      gcc@8.1.0:
        sbatch: ["-t 60", "-n 2", "-C knl"]
        module:
          swap: [PrgEnv-intel, PrgEnv-gnu/6.1.0]

bsub_compiler_group:
  type: compiler
  description: bsub example in multiple compiler groups.
  executor: local.bash
  source: src/hello.f90
  compilers:
    name: ["^(gcc|intel)"]
    default:
      gcc:
        fflags: "-O1"
        bsub: ["-W 00:30", "-n 2" ]
      intel:
        fflags: "-O2"
        bsub: ["-W 00:30", "-n 4" ]
    config:
      gcc@8.1.0:
        bsub: ["-W 00:30", "-n 6" ]
        module:
          swap: [PrgEnv-intel, PrgEnv-gnu/6.1.0]

env_example:
  type: compiler
  description: Setting environment variables
  executor: local.bash
  source: "src/hello.cpp"

```

(continues on next page)

(continued from previous page)

```

compilers:
  name: ["^(gcc)"]
  default:
    all:
      env:
        OMP_NUM_THREADS: 2
      run: $_EXEC 1 2 4
  config:
    gcc@10.2.0:
      cxxflags: "-fopenmp"

custom_env_by_compiler_group:
  type: compiler
  description: Setting environment variables in compiler groups
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: ["^(gcc|pgi)"]
    default:
      all:
        run: $_EXEC 1 2 4
      gcc:
        cxxflags: "-fopenmp"
        env:
          OMP_NUM_THREADS: 4
      pgi:
        cxxflags: "-mp"
        env:
          OMP_NUM_THREADS: 6
    config:
      gcc@10.2.0:
        env:
          OMP_NUM_THREADS: 6

      gcc@9.2.0:
        env:
          OMP_NUM_THREADS: 8

      pgi@9.2.0:
        env:
          OMP_NUM_THREADS: 10

vars_example:
  type: compiler
  description: Setting shell variables
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: ["^(gcc)"]
    default:
      all:
        vars:

```

(continues on next page)

(continued from previous page)

```

        OUTFILE: /tmp/file1.txt
        run: $_EXEC > $OUTFILE
    config:
        gcc@10.2.0:
            cxxflags: "-fopenmp"

pass_args_run:
    type: compiler
    description: Passing arguments to executable in run section
    executor: local.bash
    source: "src/hello.cpp"
    compilers:
        name: [intel]
        default:
            all:
                run: $_EXEC 1 2 4
        config:
            intel@2019:
                cxxflags: "-O1"

mpi_launcher_example:
    type: compiler
    description: mpi launcher example
    executor: local.bash
    source: "src/hello.cpp"
    compilers:
        name: [gcc]
        default:
            all:
                run: mpirun -np 2 $_EXEC
        config:
            gcc@7.3.0:
                cflags: "-O3"
                cxx: mpicxx

status_returncode_example:
    type: compiler
    description: Status returncode match example
    executor: local.bash
    source: "src/hello.cpp"
    compilers:
        name: [gnu]
        default:
            all:
                vars:
                    OUTFILE: /tmp/file1.txt
                run: $_EXEC > $OUTFILE
                status:
                    returncode: 1
        config:
            gcc@10.2.0:
                cxxflags: "-fopenmp"

```

(continues on next page)

(continued from previous page)

```

pre_post_build_run_sections:
  type: compiler
  description: Run commands pre and post build section
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: ["^(gcc)"]
    default:
      all:
        pre_build: echo "pre-build section for ALL compilers"
        post_build: echo "post-build section for ALL Compilers"
        pre_run: echo "pre-run section for ALL compilers"
        post_run: echo "post-run section for ALL Compilers"
      gcc:
        pre_build: echo "pre-build section for GCC compilers"
        post_build: echo "post-build section for GCC compilers"
        pre_run: echo "pre-run section for ALL compilers"
        post_run: echo "post-run section for ALL Compilers"
    config:
      gcc@7.3.0:
        pre_build: echo "pre-build section for gcc@7.3.0"
        post_build: echo "post-build section for gcc@7.3.0"
        pre_run: echo "pre-run section for ALL compilers"
        post_run: echo "post-run section for ALL Compilers"
        cflags: "-O3"
      gcc@8.2.0:
        pre_build: echo "gcc --version"
        cflags: "-O3"

multi_compilers:
  type: compiler
  description: Select one or more compilers to run test
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: ["^(gcc|intel|pgi|cray)"]
    exclude: [intel@18]
    default:
      gcc:
        cflags: "-fopenmp"
      intel:
        cflags: "-qopenmp"
      pgi:
        cflags: "-fopenmp"
      cray:
        cflags: "-h omp"
    config:
      gcc@7.5.0:
        cflags: "-O3"
        module:

```

(continues on next page)

(continued from previous page)

```

        load: [gcc/7.5.0]
intel@17:
  module:
    load: [intel/2017]
intel@18:
  module:
    load: [intel/2018]
pgi/18.0:
  module:
    load: [pgi/18.0]
craype/2.6.2:
  module:
    swap: [PrgEnv-intel, PrgEnv-cray]
    load: [craype/2.6.2]

metrics_example:
  type: compiler
  description: Recording test metrics with compiler schema
  executor: local.bash
  source: "src/hello.cpp"
  compilers:
    name: [gnu]
    default:
      all:
        vars:
          OUTFILE: /tmp/file1.txt
        run: $_EXEC > $OUTFILE
        status:
          returncode: 1
    config:
      gcc@10.2.0:
        cxxflags: "-fopenmp"
  metrics:
    outfile:
      vars: "OUTFILE"

```

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
 ↪ buildtest/schemas/examples/compiler-v1.0.schema.json/invalid/examples.yml

```

-----
version: "1.0"
buildspecs:
  missing_type:
    executor: local.bash
    description: "type key is missing, this is a required field"
    source: "src/hello.c"
    compilers:
      name: [intel]

  missing_required_compilers:
    executor: local.bash
    type: compiler
    description: "missing required field compilers "

```

(continues on next page)

(continued from previous page)

```

source: "src/hello.c"

missing_required_source:
  executor: local.bash
  type: compiler
  description: "missing required field 'source' "
  compilers:
    name: [gcc]

invalid_type_value:
  executor: local.bash
  type: script
  description: "invalid value for type field must be 'compiler' "
  source: src/hello.c
  compilers:
    name: [gcc]

invalid_description_value:
  executor: local.bash
  type: compiler
  description: 1
  source: src/hello.c
  compilers:
    name: [gcc]

invalid_type_module:
  executor: local.bash
  type: compiler
  description: "type for 'module' key, expecting a property but received 'string' "
  source: src/hello.c
  compilers:
    name: [gcc]
    config:
      gcc/9.2.0:
        module: "module load gcc/9.2.0"

module_purge_invalid_type:
  executor: local.bash
  type: compiler
  description: "The purge property module is invalid. Expects bool got an int"
  source: src/hello.c
  compilers:
    name: [gcc]
    config:
      gcc/9.2.0:
        module:
          purge: 1

module_swap_duplicate_check:
  executor: local.bash
  type: compiler
  description: "The swap property expects two unique items"

```

(continues on next page)

(continued from previous page)

```
source: src/hello.c
compilers:
  name: [gcc]
  config:
    gcc/9.2.0:
      module:
        swap: [gcc/8.0, gcc/8.0]

module_swap_min_items:
  executor: local.bash
  type: compiler
  description: "The swap property expects a minimum of 2 items"
  source: src/hello.c
  compilers:
    name: [gcc]
    config:
      gcc/9.2.0:
        module:
          swap: [gcc/8.0]

module_swap_max_items:
  executor: local.bash
  type: compiler
  description: "The swap property expects a maximum of 2 items"
  source: src/hello.c
  compilers:
    name: [gcc]
    config:
      gcc/9.2.0:
        module:
          swap: [gcc/8.0, gcc/9.0, gcc/10.0]

module_load_duplicate_items:
  executor: local.bash
  type: compiler
  description: "The load property expects unique items"
  source: src/hello.c
  compilers:
    name: [gcc]
    config:
      gcc/9.2.0:
        module:
          load: [gcc/9.2.0, gcc/9.2.0]

module_load_min_items:
  executor: local.bash
  type: compiler
  description: "The load property expects a minimum of 1 item"
  source: src/hello.c
  compilers:
    name: [gcc]
    config:
```

(continues on next page)

(continued from previous page)

```

    gcc/9.2.0:
      module:
        load: []

additionalProperties_main:
  executor: local.bash
  type: compiler
  description: "test additionalProperties in main schema"
  foo: bar
  source: src/hello.c
  compilers:
    name: [gcc]

missing_required_compiler_name:
  executor: local.bash
  type: compiler
  description: "'name' field in compilers section is required field"
  source: src/hello.f90
  compilers:
  default:
    cray:
      fflags: "-O1"
  config:
    PrgEnv-cray@2.6.2:
      module:
        swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

uniqueItems_compiler_name:
  executor: local.bash
  type: compiler
  description: "Test unique items in 'name' field in compilers section"
  source: src/hello.f90
  compilers:
    name: ["^(PrgEnv-cray)", "^(PrgEnv-cray)"]
  config:
    PrgEnv-cray@2.6.2:
      fflags: "-O1"
      module:
        swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

additionalProperties_compiler:
  executor: local.bash
  type: compiler
  description: "Test additionalProperties in compiler section"
  source: src/hello.f90
  compilers:
    name: ["PrgEnv-cray"]
    FOO: BAR
  default:
    all:
      env:
        X: 1

```

(continues on next page)

(continued from previous page)

```

config:
  PrgEnv-cray@2.6.2:
    fflags: "-O1"
    module:
      swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

additionalProperties_compiler_default_all:
  executor: local.bash
  type: compiler
  description: "Test additionalProperties in compiler default all section"
  source: src/hello.f90
  compilers:
    name: ["PrgEnv-cray"]
    default:
      all:
        XYZ: 123
  config:
    PrgEnv-cray@2.6.2:
      fflags: "-O1"
      module:
        swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

additionalProperties_compiler_config:
  executor: local.bash
  type: compiler
  description: "Test additionalProperties in compiler config section, FOO: BAR"
  source: src/hello.f90
  compilers:
    name: ["PrgEnv-cray"]
    config:
      PrgEnv-cray@2.6.2:
        FOO: BAR
        fflags: "-O1"
        module:
          swap: [PrgEnv-intel, PrgEnv-cray/2.6.2]

```

Spack Schema Examples

```

$ buildtest schema -n spack-v1.0.schema.json --example
File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/
↳ buildtest/schemas/examples/spack-v1.0.schema.json/valid/examples.yml
-----
version: "1.0"
buildspecs:
  env_create_name:
    type: spack
    executor: generic.local.sh
    description: Create spack environment by name
  spack:
    root: $HOME/spack/

```

(continues on next page)

(continued from previous page)

```
env:
  create:
    name: myproject
  specs:
    - m4
    - zlib
  install:
    options: ''

env_activate:
  type: spack
  executor: generic.local.sh
  description: Activate spack environment by name
  spack:
    root: $HOME/spack/
    env:
      activate:
        name: myproject
      specs:
        - m4
        - zlib
    install:
      options: ''

env_create_directory:
  type: spack
  executor: generic.local.sh
  description: Create spack environment by directory
  spack:
    root: $HOME/spack/
    env:
      create:
        dir: $HOME/spack-env/myproject
      specs:
        - 'm4'
        - 'zlib@1.2.11'
    install:
      options: '--cache-only'

env_create_from_manifest:
  type: spack
  executor: generic.local.sh
  description: Create spack environment from manifest file
  spack:
    root: $HOME/spack/
    env:
      create:
        name: myproject
        manifest: $HOME/spack.yaml
    install:
      options: '--cache-only'
```

(continues on next page)

(continued from previous page)

```

env_concretized_install:
  type: spack
  description: run 'spack concretize -f' in an environment and install specs
  executor: generic.local.sh
  spack:
    root: $HOME/spack/
    env:
      create:
        name: myproject
        manifest: $HOME/spack.yaml
        concretize: true
    install:
      options: '--cache-only'

env_mirror:
  type: spack
  executor: generic.local.sh
  description: declare spack mirror 'spack mirror add h5 /path/to/mirror' in
↳ environment
  spack:
    root: $HOME/spack/
    env:
      mirror:
        h5: /path/to/mirror
      create:
        name: myproject
        manifest: $HOME/spack.yaml
    install:
      options: '--cache-only'

install_without_env:
  type: spack
  executor: generic.local.sh
  description: install specs without environment
  spack:
    root: $HOME/spack/
    install:
      options: '--cache-only'
      specs: ['m4', 'bzip2']

pre_post_cmd_spack_install:
  type: spack
  executor: generic.local.sh
  description: run commands before and after spack using pre_cmds and post_cmds
  pre_cmds: |
    cd $HOME
    git clone https://github.com/spack/spack
  spack:
    root: $HOME/spack/
    install:
      options: '--cache-only'

```

(continues on next page)

(continued from previous page)

```

    specs: ['m4', 'bzip2']
post_cmds: |
    spack find

remove_spack_environment:
    type: spack
    executor: generic.local.sh
    description: remove spack environment explicitly before creating environment
    spack:
        root: $HOME/spack
        env:
            rm:
                name: m4
            create:
                name: m4
            activate:
                name: m4
            specs:
                - 'm4'

remove_spack_environment_automatically:
    type: spack
    executor: generic.local.sh
    description: remove spack environment automatically
    spack:
        root: $HOME/spack
        env:
            create:
                remove_environment: true
                name: m4
            activate:
                name: m4
            specs:
                - 'm4'

spack_test_results_specs_suites:
    type: spack
    executor: generic.local.sh
    description: "Report results using suite name and spec format in spack test results"
    tags: [spack]
    pre_cmds: |
        cd /tmp
        git clone https://github.com/spack/spack
    spack:
        root: /tmp/spack
        verify_spack: false
        install:
            specs: ['m4', 'zlib' ]
        test:
            remove_tests: true
        run:

```

(continues on next page)

(continued from previous page)

```

        specs: ['m4', 'zlib']
    results:
        option: '-f'
        suite: ['zlib']
        specs: ['m4']

    post_cmds: |
        spack find
        rm -rf $SPACK_ROOT

    sbatch_field_with_spack:
        type: spack
        executor: generic.local.sh
        description: Specify sbatch field with spack schema
        sbatch: ["-N1", "-q normal", "-t 10", "-M 30M"]
        spack:
            root: $HOME/spack
            install:
                specs: ['m4']
                options: "-v"

    skip_test_in_spack:
        skip: True
        type: spack
        executor: generic.local.sh
        description: This test will be skipped
        spack:
            root: $HOME/spack
            install:
                specs: ['m4']
                options: "-v"

    var_declaration_in_spack:
        type: spack
        executor: generic.local.sh
        description: Define variables and environment variables in spack
        vars:
            FOO: "BAR"
        env:
            SPACK_GNUPGHOME: "$HOME/.gnupg"
        spack:
            root: $HOME/spack
            install:
                specs: ['m4']
                options: "-v"

    metrics_example_vars_envs:
        type: spack
        executor: generic.local.sh
        description: Define metrics variable and environment variable in spack
        vars:
            FOO: "BAR"

```

(continues on next page)

(continued from previous page)

```

env:
  SPACK_GNUPGHOME: "$HOME/.gnupg"
metrics:
  foo:
    vars: "FOO"
  spack_gpg:
    env: "SPACK_GNUPGHOME"
spack:
  root: $HOME/spack
  install:
    specs: ['m4']
    options: "-v"

spack_sbbatch_multi_executors:
  type: spack
  executor: "generic.local.(sh|bash)"
  description: "sbbatch directives can be defined in spack schema"
  tags: [spack]
  executors:
    generic.local.sh:
      sbbatch: ["-N 1", "-t 20"]
    generic.local.bash:
      sbbatch: ["-N 8", "-t 10"]
  spack:
    root: $HOME/spack
    env:
      specs:
        - 'm4'
    activate:
      name: m4
    concretize: true

vars_multi_executors:
  type: spack
  executor: "generic.local.(sh|bash)"
  description: "variable declaration with multiple executors"
  tags: [spack]
  executors:
    generic.local.sh:
      vars:
        FOO: BAR
    generic.local.bash:
      vars:
        HELLO: WORLD
  spack:
    root: $HOME/spack
    env:
      specs:
        - 'm4'
    activate:
      name: m4
    concretize: true

```

(continues on next page)

(continued from previous page)

File: /home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/

↪ buildtest/schemas/examples/spack-v1.0.schema.json/invalid/examples.yml

version: "1.0"

buildspecs:

missing_root_to_spack:

type: spack

executor: generic.local.sh

description: 'root property required to run source \$SPACK_ROOT/share/spack/setup-env.

↪ sh'

spack:

env:

create:

name: myproject

specs:

- m4

- zlib

specs_must_be_list_of_strings:

type: spack

executor: generic.local.sh

description: 'specs must be a list of strings '

spack:

root: \$HOME/spack

env:

create:

name: myproject

specs:

- 1

- zlib

additionalProperties_spack_field:

type: spack

executor: generic.local.sh

description: additional Properties can't be specified in spack section

spack:

root: \$HOME/spack

FOO: BAR

env:

create:

name: myproject

specs:

- zlib

invalid_type_mirror_field:

type: spack

executor: generic.local.sh

description: The mirror field must be a key value pair

spack:

root: \$HOME/spack

mirror: https://caches.e4s.io

env:

(continues on next page)

(continued from previous page)

```

    create:
      name: myproject
    specs:
      - zlib

spack_test_additionalProperties:
  type: spack
  executor: generic.local.sh
  description: "Check for additionalProperties in test section. F00 key is not allowed"
  tags: [spack]
  pre_cmds: |
    cd /tmp
    git clone https://github.com/spack/spack
  spack:
    root: /tmp/spack
    verify_spack: false
    install:
      specs: ['m4', 'zlib' ]
    test:
      F00: BAR
      remove_tests: true
      run:
        specs: ['m4', 'zlib']

spack_test_results_testing_required_oneOf:
  type: spack
  executor: generic.local.sh
  description: "The results property expects one to specify specs, suite, or both"
  tags: [spack]
  pre_cmds: |
    cd /tmp
    git clone https://github.com/spack/spack
  spack:
    root: /tmp/spack
    verify_spack: false
    install:
      specs: ['m4', 'zlib' ]
    test:
      remove_tests: true
      run:
        specs: ['m4', 'zlib']
      results:
        option: '-f'
  post_cmds: |
    spack find
    rm -rf $SPACK_ROOT

spack_test_run_invalid_spec:
  type: spack
  executor: generic.local.sh
  description: "specs property requires a list of strings. "
  tags: [spack]

```

(continues on next page)

(continued from previous page)

```
pre_cmds: |
    cd /tmp
    git clone https://github.com/spack/spack
spack:
    root: /tmp/spack
    verify_spack: false
    install:
        specs: ['m4', 'zlib' ]
    test:
        remove_tests: true
        run:
            specs: ['m4', 1]
        results:
            option: '-f'
            suite: ['zlib']
            specs: ['m4']

post_cmds: |
    spack find
    rm -rf $SPACK_ROOT
```

3.10 Using buildtest at HPC sites

We assume you have read the *Buildtest Tutorial* and *Configuring buildtest* and now you want to use buildtest at your site. This document will highlight some points to consider before you start.

To get started, you should consider standing up an empty repository where you will host your tests. This can be GitHub, GitLab, bitbucket, etc...

3.10.1 Picking a version of buildtest

If you are going to use buildtest, you should consider if you want to use the bleeding edge (*devel*), stable release (*master*) or a *tag release*. Generally, we recommend you start off with stable release and then incrementally update your buildtest with new *releases* as they come out and check the *CHANGELOG.rst* for updates between version release.

Please make sure to read the appropriate version documentation based on the version of buildtest.

- Devel Docs: <https://buildtest.readthedocs.io/en/devel/index.html>
- Stable Docs: <https://buildtest.readthedocs.io/en/latest/>

3.10.2 Configuring buildtest for your site

Once you have picked a version of buildtest, you need to configure buildtest for your site, this requires you see *Configuring buildtest*. We recommend you see *buildtest-cori configuration* that provides how buildtest is configured at NERSC. Once you have defined your configuration file you should make sure your configuration is valid by running:

```
buildtest config validate
```

3.10.3 Writing Test

If you are going to write test, we assume you have read *Buildspec Tutorial* section which covers how to write buildsspecs. You should consider reviewing the Schema Documentation: <https://buildtesters.github.io/buildtest/> which goes in detail about each schema and buildspec attributes.

If you are writing tests, it's generally good practice to *define tags* in your test so you can group tests by a tagname and run them via `buildtest build --tags`. If you plan to use tags to run your tests, you should document tags and how they are meant to be used.

3.11 Conference and Publications

3.11.1 Talks

Conference	Date	Link
Facility Testing of E4S via E4S Testsuite, Spack Test, and buildtest	Sep 14, 2021	TBD
ECP Annual Meeting 2021	Apr 15, 2021	PDF, VIDEO
High Performance Computing & Simulation 2020	Mar 26, 2021	PDF
SEA Improving Scientific Software 2021	Mar 23, 2021	PDF, VIDEO
FOSDEM21	Feb 7, 2021	PDF
6th Easybuild User Meeting	Jan 29, 2021	PDF, VIDEO
FOSDEM20	Feb 2, 2020	PDF, VIDEO
5th Easybuild User Meeting	Jan 30, 2020	PDF, VIDEO
SC19 @ HUST workshop	Nov 18, 2019	PDF
HPCKP'18	June 22, 2018	PDF
HPCKP'17	June 15, 2017	PDF

3.11.2 Publications

- Siddiqui S. (2020) Buildtest: A Software Testing Framework with Module Operations for HPC Systems . In: Juckeland G., Chandrasekaran S. (eds) Tools and Techniques for High Performance Computing. HUST 2019, SE-HER 2019, WIHPC 2019. Communications in Computer and Information Science, vol 1190. Springer, Cham

3.11.3 Article

- <https://www.hpcwire.com/2019/01/17/pfizer-hpc-engineer-aims-to-automate-software-stack-testing/>

3.12 Contributing Guide

This guide is geared for developers and maintainers of buildtest who want to contribute back to buildtest project. There are many ways you can help contribute to buildtest that may include:

- Improve user documentation
- Increase test coverage of buildtest regression tests.
- Work on an [existing issue](#)
- Report a bug or new feature requests at <https://github.com/buildtesters/buildtest/issues>

3.12.1 Overview

buildtest codebase is written in Python 3, so if you are new to Python you will want to check out the python 3 tutorial: <https://docs.python.org/3/tutorial/>. This is a good starting point to understand python basics. If you are familiar with Python 2 you may want to review the [Python 2-3 cheat sheet](#).

buildtest relies on [YAML](#) and [JSON Schema](#), you should review [Understanding JSON Schema](#) article as it provides a thorough overview of JSON Schema. There are several resources to help you learn YAML for instance you can check out:

- <https://www.tutorialspoint.com/yaml/index.htm>
- <https://learnxinyminutes.com/docs/yaml/>

buildtest has a regression test that is run via [pytest](#). You should be familiar with pytest and it's usage and documentation as it will help you write regression test. The regression test makes use of [coverage](#) to measure code coverage of buildtest source code. This is configured using [.coveragerc](#) file located in top of repo. The coverage data is pushed to [codecov](#) at <https://codecov.io/gh/buildtesters/buildtest/>.

buildtest has several CI checks written in GitHub workflows. These are found in [.github/workflows](#) directory of buildtest. You should familiarize yourself with [github workflow syntax](#) if you want to contribute back to github workflows.

Git is essential to code contribution so we recommend you get comfortable using *git* as it will be discussed in [code contributing guide](#). We recommend you review one the following guides to help you learn *git*:

- <https://guides.github.com/introduction/git-handbook/>
- <https://git-scm.com/docs/gittutorial>
- <https://guides.github.com/>
- <https://lab.github.com/>

buildtest documentation is built on [sphinx](#) and hosted via [readthedocs](#). Be sure to check out [documentation on readthedocs](#) to understand how it works. The buildtest project is hosted at <https://readthedocs.org/projects/buildtest/> which hosts the public documentation at <https://buildtest.readthedocs.io/>. The documentation pages are written in [reStructured Text \(rST\)](#) which is Sphinx's markup language when hosting the docs.

3.12.2 Contributing Topics

Code Contribution Guide

This guide will walk through the code contribution guide, we expect you have a *github account* and experience using *git* and familiarity with GitHub interface.

GitHub Account

If you don't have a GitHub account please [register](#) your account.

Fork the repo

First, you'll need to fork the repo <https://github.com/buildtesters/buildtest>

You might need to setup your SSH keys in your git profile if you are using ssh option for cloning. For more details on setting up SSH keys in your profile, follow instruction found in <https://help.github.com/articles/connecting-to-github-with-ssh/>

SSH key will help you pull and push to repository without requesting for password for every commit. Once you have forked the repo, clone your local repo:

```
git clone git@github.com:YOUR\_GITHUB\_LOGIN/buildtest.git
```

Adding Upstream Remote

First you need to add the upstream repo, to do this you can issue the following:

```
git remote add upstream git@github.com/buildtesters/buildtest.git
```

The upstream tag is used to sync changes from upstream repo to keep your repo in sync before you contribute back.

Make sure you have set your user name and email set properly in git configuration. We don't want commits from unknown users. This can be done by setting the following:

```
git config user.name "First Last"
git config user.email "abc@example.com"
```

For more details see [First Time Git Setup](#)

Sync your branch from upstream

The devel from upstream will get Pull Requests from other contributors, in-order to sync your forked repo with upstream, run the commands below:

```
git checkout devel
git fetch upstream devel
git pull upstream devel
```

Once the changes are pulled locally you can sync devel branch with your fork as follows:

```
git checkout devel
git push origin devel
```

Repeat this same operation with `master` branch if you want to sync it with upstream repo

Contribution Workflow

If you want to contribute back, you should create a feature branch from *devel* and add your files, commit and push them to your fork. The workflow can be summarized as follows:

```
git checkout devel
git checkout -b featureX
git add <file1> <file2> ...
git commit -m "commit message"
git push origin featureX
```

Once the branch is created in your fork, you can [create a Pull Request](https://github.com/buildtesters/buildtest) with the destination branch `devel` at <https://github.com/buildtesters/buildtest> and base branch which is your feature branch pushed at your fork.

Note: Do not push to `master` or `devel` branch on your fork or upstream.

Pull Request Review

Once you have submitted a Pull Request, please check the automated checks that are run for your PR to ensure checks are passed. Most common failures in CI checks are black and pyflakes issue, this can be done by [configuring black](#) and running [pyflakes](#). Once all checks have passed, maintainer will review your PR and provide feedback so please be patient. Please coordinate with maintainer through PR or Slack.

Resolving PR Merge Conflicts

Often times, you may start a feature branch and your PR get's out of sync with `devel` branch which may lead to conflicts, this is a result of merging incoming PRs that may cause upstream *HEAD* to change over time which can cause merge conflicts. This may be confusing at first, but don't worry we are here to help. For more details about merge conflicts click [here](#).

Syncing your feature branch with *devel* is out of scope for this documentation, however you can use the steps below as a *guide* when you run into this issue.

You may want to take the steps to first sync `devel` branch and then selectively rebase or merge `devel` into your feature branch.

First go to `devel` branch and fetch changes from upstream:

```
git checkout devel
git fetch upstream devel
```

Note you shouldn't be making any changes to your local `devel` branch, if `git fetch` was successful you can merge your `devel` with upstream as follows:

```
git merge upstream/devel
```

Next, navigate to your feature branch and sync feature changes with devel:

```
git checkout <feature-branch>
git merge devel
```

Note: Running above command will sync your feature branch with devel but you may have some file conflicts depending on files changed during PR. You will need to resolve them manually before pushing your changes

Instead of merge from devel you can rebase your commits interactively when syncing with devel. This can be done by running:

```
git rebase -i devel
```

Once you have synced your branch push your changes and check if file conflicts are resolved in your Pull Request:

```
git push origin <feature-branch>
```

General Tips

1. It's good practice to link PR to an issue during commit message. Such as stating `Fix #132` for fixing issue 132.
2. If you have an issue, ask your question in slack before reporting issue. If your issue is not resolved check any open issues for resolution before creating a new issue.
3. For new features or significant code refactor please notify maintainers and open an issue before working on task to keep everyone informed.
4. If you open an issue, please respond back during discussion, if there is no activity the issue will be closed.
5. Please refrain from opening duplicate issue, check if there is an existing issue addressing similar problem, instead you can participate in discussion in the issue or contact appropriate individuals directly in slack.
6. There should not be any branches other than master or devel. Feature branches should be pushed to your fork and not to origin.

Configuring Black Pre-Commit Hook

To configure pre-commit hook, make sure you install [pre-commit](#) via `pip install pre-commit`. The *pre-commit* utility should be available if you install extra dependencies from buildtest (`pip install -r docs/requirements.txt`).

You can configure `.pre-commit-config.yaml` with the version of python you are using. It is currently setup to run for python 3.7 version as follows:

```
language_version: python3.7
```

Alter this value based on python version you are using or refer to [black version control integration](#).

To install the pre-commit hook run:

```
$ pre-commit install
pre-commit installed at .git/hooks/pre-commit
```

This will invoke hook `.git/hooks/pre-commit` prior to `git commit`. Shown below we attempt to commit which resulted in pre commit hook and caused black to format code.

```
$ git commit -m "test black commit with precommit"
black.....Failed
- hook id: black
- files were modified by this hook

reformatted buildtest/config.py
All done!
1 file reformatted.
```

If you are interested in running black locally to see diff result from black without auto-formatting code, you can do the following:

```
$ black --check --diff .
--- tests/test_inspect.py      2020-02-25 18:58:58.360360 +0000
+++ tests/test_inspect.py      2020-02-25 18:59:07.336414 +0000
@@ -18,11 +18,11 @@
 def test_distro_short():
     assert "rhel" == distro_short("Red Hat Enterprise Linux Server")
     assert "centos" == distro_short("CentOS")
     assert "suse" == distro_short("SUSE Linux Enterprise Server")
-    x=0+1*3
+    x = 0 + 1 * 3
```

The changes will be shown with lines removed or added via - and +. For more details refer to [black documentation](#).

isort

isort is a python utility that will sort python imports alphabetically. We use isort as part of the CI checks, there is a [.isort.cfg](#) that defines the isort configuration that is compatible with **black** utility. We have setup a pre-commit hook that can be used to automatically run isort as part of your `git commit` process. This is defined in pre-commit configuration file [.pre-commit-config.yaml](#) that can be installed by running `pre-commit install`. Once this is setup, you will see **isort** and **black** checks are run during the commit process.

```
$ git commit
isort.....Passed
black.....Passed
[sphinx_fix 85d9d42c] fix issue with rendering bullet points in sphinx. This is solved_
↳by downgrading docutils to version 0.16.
2 files changed, 5 insertions(+)
```

Please make sure you run `pip install -r docs/requirements.txt` to get the development dependencies that includes isort.

If you want to run isort, you can use the `-c` and `-diff` option to check and see diff between files. For instance in example below we see isort reports changes to import statement

```
$ isort -c --diff profile black buildtest/main.py
ERROR: /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/main.py Imports are_
↳incorrectly sorted and/or formatted.
--- /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/main.py:before      2021-
↳07-13 16:53:42.722718
+++ /Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/main.py:after      2021-
↳07-13 16:54:12.135986
```

(continues on next page)

(continued from previous page)

```

@@ -1,8 +1,7 @@
"""Entry point for buildtest"""

+import os
  import webbrowser
-import os
-

  from buildtest.cli import get_parser
  from buildtest.cli.build import BuildTest
Broken 2 paths

```

If you want to apply the changes you can get rid of `-c` and `--diff` option and `isort` will apply the changes. Please see https://pycqa.github.io/isort/docs/configuration/black_compatibility.html and https://black.readthedocs.io/en/stable/guides/using_black_with_other_tools.html#isort for documentation regarding black and isort compatibility.

pyflakes

`pyflakes` is a program that checks for python source code for errors such as unused imports. We have configured an automated check to test your incoming PR using `pyflakes`. `pyflakes` should be available in your python environment if you installed buildtest extra dependencies in `requirements.txt` (`pip install -r docs/requirements.txt`).

You can run `pyflakes` against any file or directory the ones of importance is running `pyflakes` against buildtest source code and regression test. You can do that by running:

```
pyflakes buildtest tests
```

GitHub Integrations

buildtest has several CI checks that are run when you create a Pull Request, it is your responsibility to review the CI checks and make sure all checks are passing. Each pull request will show the CI checks, you can see the [github actions](#) that are also typically linked as part of the pull request.

Coverage

We use `coverage` to measure code coverage of buildtest when running regression test. We use CodeCov to display coverage reports through web interface. The coverage configuration is managed by `.coveragerc` file found in the root of the repo.

Whenever you add new feature to buildtest, please add regression test with test coverage to help maintainers review new feature request. For more details on running coverage tests see [Running test via coverage](#).

CodeCov

Codecov report coverage details in web-browser. CodeCov can perform [pull request comments](#) after coverage report is uploaded to Codecov which is useful for reviewer and assignee to see status of coverage report during PR review process. The codecov file `.codecov.yml` is used for configuration codecov. For more details on codecov yaml file see <https://docs.codecov.io/docs/codecov-yaml>.

Gitlab CI checks

buildtest has automated CI checks on gitlab servers: <https://software.nersc.gov> and <https://code.ornl.gov>. The gitlab pipelines are stored in `.gitlab` directory found in root of repository.

The `mirror.yml` github workflow is responsible for mirroring and trigger CI check and return result back to github PR. Currently, we are using github action [stenongithub/gitlab-mirror-and-ci-action](#) to perform pull mirroring and triggering CI job.

The gitlab server <https://software.nersc.gov> is hosted at NERSC. The following steps were taken to setup pipeline

1. Create a Personal Access token with **read_api**, **read_repository**, **write_repository** scope at https://software.nersc.gov/-/profile/personal_access_tokens
2. Define a secret **CORI_GITLAB_PASSWORD** at <https://github.com/buildtesters/buildtest/settings/secrets/actions> with token value generated in step 1
3. Import buildtest project from github at <https://software.nersc.gov/siddiq90/buildtest>
4. Add variable **SECRET_CODECOV_TOKEN** in https://software.nersc.gov/siddiq90/buildtest/-/settings/ci_cd that contains codecov token found at <https://app.codecov.io/gh/buildtesters/buildtest/settings>
5. Change gitlab CI configuration file to `.gitlab/cori.yml` under **Settings > CI/CD > General pipelines**. For more details see <https://docs.gitlab.com/ee/ci/pipelines/settings.html#custom-cicd-configuration-path>

The gitlab server <https://code.ornl.gov> is hosted at OLCF which has access to systems like Summit and Ascent. We performed similar steps as shown above with slight modification

1. Create a Personal access token with same scope at https://code.ornl.gov/-/profile/personal_access_tokens
2. Define a secret **OLCF_GITLAB_PASSWORD** at <https://github.com/buildtesters/buildtest/settings/secrets/actions>
3. Import buildtest project at <https://code.ornl.gov/ecpcitest/buildtest>. Currently, all projects in `ecpcitest` project group has access to gitlab runners.
4. Add variable **SECRET_CODECOV_TOKEN** in https://code.ornl.gov/ecpcitest/buildtest/-/settings/ci_cd that contains codecov token found at <https://app.codecov.io/gh/buildtesters/buildtest/settings>
5. Change gitlab CI configuration file to `.gitlab/olcf.yml`

Currently, the gitlab pipelines are triggered manually which requires a user to have access to the gitlab project to run the pipeline. The pipelines can be run manually at <https://software.nersc.gov/siddiq90/buildtest/-/pipelines> and <https://code.ornl.gov/ecpcitest/buildtest/-/pipelines>

The github workflow `mirror.yml` defines gitlab configuration for each mirror. Any changes to mirror path must be addressed in this workflow to ensure pull mirroring is done properly.

GitHub Bots

buildtest has a few bots to do various operations that are described below.

- **Stale** - stale bot is used to close outdated issues. This is configured in `.github/stale.yml`. If there is no activity on a issue after certain time period, **probot-stale** will mark the issue and project maintainers can close it manually. For more details on Stale refer to the [documentation](#)
- **CodeCov** - The codecov bot will report codecov report from the issued pull request once coverage report is complete. The configuration for codecov is defined in `.codecov.yml` found in root of repo.
- **Pull Request Size** - is a bot that labels Pull Request by number of **changed** lines of code.

Building Documentation

The buildtest documentation is written in [reStructuredText](#) using sphinx. You should be familiar with rst if you want to contribute to user documentation.

ReadTheDocs

buildtest [documentation](#) is hosted by ReadTheDocs at <https://readthedocs.org> which is a documentation platform for building and hosting your docs.

buildtest project can be found at <https://readthedocs.org/projects/buildtest/> which will show the recent builds and project setting. If you are interested in becoming a maintainer, please contact **Shahzeb Siddiqui** (shahzebmsiddiqui@gmail.com) to grant access to this project.

Setup

buildtest documentation is located in top-level [docs](#) directory. If you want to build the documentation you will need to make sure your python environment has all the packages defined in `docs/requirements.txt`. If your environment is already setup as described in [Installing buildtest](#) then you can skip this step.

To install your python packages, you can run the following:

```
pip install -r docs/requirements.txt
```

Building docs locally

To build your documentation, navigate to the `docs` directory and run the following:

```
cd docs
make clean
make html
```

It is best practice to run `make clean` to ensure sphinx will remove old html content from previous builds, but it is ok to skip this step if you are making minor changes.

Running `make html` will build the sphinx project and generate all the html files in `docs/_build/html`. Once this process is complete you can view the html pages by running the following:

```
open _build/html/index.html
```

Please refer to the `Makefile` to see list of tags or run `make` for additional help.

Sphinx

The documentation is built via `Sphinx` using `reStructuredText (rST)` as its markup language. When you run `make` you are running `sphinx-build` command which will generate the documentation.

Sphinx will read the configuration file `conf.py` used for building the project. We have enabled a couple `sphinx extensions` in our project to customize our documentation

API Generation

We make use of `Sphinx AutoAPI` to generate buildtest API documentation that is hosted on <https://buildtest.readthedocs.io/en/devel/api/index.html>. The Sphinx AutoAPI configuration is configured in sphinx configuration file `conf.py`. For more details on configuration options see <https://sphinx-autoapi.readthedocs.io/en/latest/reference/config.html>

Command Line Documentation

We make use of `sphinx-argparse` to generate documentation for buildtest command line that is hosted at <https://buildtest.readthedocs.io/en/devel/command.html>. In order to use this tool one must install this package and enable the extension in sphinx configuration.

DocStrings

We have enabled `napoleon extension` to support Google style docstring. Please follow this format when you are writing docstring for buildtest codebase. For more details on google style see: <https://google.github.io/styleguide/pyguide.html>

Automate Documentation Examples

buildtest has a script in top-level folder `script/docgen.py` to automate documentation examples. This script can be run as follows:

```
python script/docgen.py
```

This assumes your buildtest environment is setup, the script will write documentation test examples in `docs/docgen`. Consider running this script when **adding**, **modifying**, or **removing** documentation examples. Once the test are complete, you will want to add the tests, commit and push as follows:

```
git add docs/docgen
git commit -m <MESSAGE>
git push
```

Regression Tests

buildtest has a suite of regression tests to verify the state of buildtest. These tests are located in the top-level directory `tests`. buildtest is using `pytest` for running the regression tests.

Getting Started

In order to write regression tests, you should have `pytest` and `coverage` installed in your python environment. You can do this by installing all dependencies found in requirements file:

```
pip install -r docs/requirements.txt
```

Writing Regression Tests

If you want to write a new regression test, you should be familiar with `coverage` report that is pushed to `codecov`. The coverage report will give a detailed line-line coverage of source code HIT/MISS when running the regression test. Increasing coverage report would be great way to write a new regression test.

The `tests` directory is structured in a way that each source file has a corresponding test file that starts with `test_`. For instance, if you want to write a test for `buildtest/utils/command.py`, there will be a corresponding test under `tests/utils/test_command.py`.

If you adding a new directory, make sure the name corresponds to one found under `buildtest` directory and add a `__init__.py` in the new directory. This is required by `pytest` for test discovery. All test methods must start with `test_` in order for `pytest` to run your regression test.

Shown below is a simple test that always passes

```
def test_regression_example1():
    assert True
```

For more details on writing tests with `pytest` see [Getting-Started](#).

Running Regression Test

The recommended way to run regression test is via:

```
$ python $BUILDTEST_ROOT/scripts/regtest.py
```

This script is a wrapper to `pytest` and `coverage`. We have a `pytest.ini` found in top-level folder that defines `pytest` configuration. If you want to run tests natively via `pytest` without using the script you can just run `pytest` and it will run with options defined in `pytest.ini` file.

If you want to run all schema tests you can use the `schema` marker as follows:

```
pytest -v -m schema
```

To see a list of `pytest` markers see `pytest.ini` or run:

```
pytest --markers
```

For a complete list of options refer to `pytest` [documentation](#) or run `pytest --help`.

Running test via coverage

There is a coverage configuration file `.coveragerc` located in root of buildtest that is read by **coverage** utility. The `regtest.py` script will collect coverage details upon completion of regression test which is equivalent to running `coverage run -m pytest` but we make some additional checks when running the script. Upon completion of tests you can run `coverage report` to show coverage results of your regression test run locally. Shown below is an example output:

```
$ coverage report
```

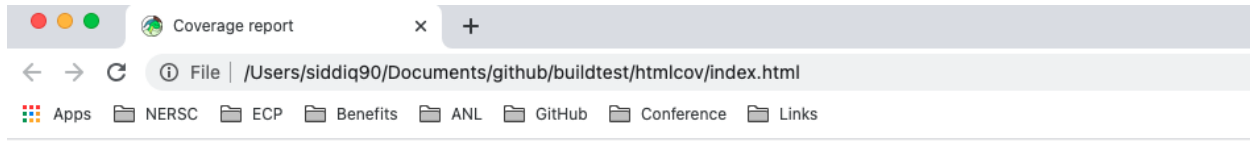
Name	Stmts	Miss	Branch	BrPart	Cover

buildtest/__init__.py	3	3	0	0	0.00%
buildtest/defaults.py	17	17	0	0	0.00%
buildtest/executors/slurm.py	110	93	28	0	12.32%
buildtest/executors/cobalt.py	110	93	22	0	12.88%
buildtest/executors/pbs.py	96	81	14	0	13.64%
buildtest/executors/lsf.py	103	85	16	0	15.13%
buildtest/utils/timer.py	15	9	4	0	31.58%
buildtest/menu/__init__.py	29	16	10	0	33.33%
buildtest/executors/setup.py	108	60	60	8	35.71%
buildtest/menu/compiler.py	107	60	50	3	38.22%
buildtest/config.py	158	72	76	10	47.86%
buildtest/system.py	155	70	38	11	50.78%
buildtest/docs.py	5	2	0	0	60.00%
buildtest/log.py	19	7	0	0	63.16%
buildtest/buildsystem/base.py	185	45	50	8	67.23%
buildtest/menu/build.py	421	117	208	22	70.59%
buildtest/buildsystem/batch.py	75	17	44	7	71.43%
buildtest/buildsystem/compilerbuilder.py	193	36	52	10	77.14%
buildtest/buildsystem/builders.py	107	24	60	8	77.25%
buildtest/utils/tools.py	19	2	12	2	80.65%
buildtest/exceptions.py	7	2	4	0	81.82%
buildtest/menu/buildspec.py	356	46	188	22	83.82%
buildtest/executors/local.py	49	3	10	4	88.14%
buildtest/buildsystem/scriptbuilder.py	41	3	10	3	88.24%
buildtest/utils/file.py	68	11	30	0	88.78%
buildtest/menu/report.py	193	16	114	14	89.58%
buildtest/executors/base.py	61	4	18	2	89.87%
buildtest/utils/command.py	68	3	20	5	90.91%
buildtest/menu/config.py	73	4	12	1	94.12%
buildtest/buildsystem/parser.py	51	2	18	2	94.20%
buildtest/menu/schema.py	26	0	16	2	95.24%
buildtest/menu/inspect.py	63	2	46	3	95.41%
buildtest/schemas/defaults.py	32	0	0	0	100.00%
buildtest/schemas/utils.py	26	0	8	0	100.00%
buildtest/utils/shell.py	30	0	8	0	100.00%

TOTAL	3179	1005	1246	147	66.19%

4 empty files skipped.

If you want to view the coverage details locally in a browser you can run: `coverage html` which will write the results to directory `htmlcov`. You can open the file `open htmlcov/index.html` and it will show you a summary of coverage results that you would see from codecov. Shown below is a preview of coverage report that you would see after running your regression test.



Coverage report: 66.19%

Module ↑	statements	missing	excluded	branches	partial	coverage
buildtest/__init__.py	3	3	0	0	0	0.00%
buildtest/buildsystem/base.py	185	45	4	50	8	67.23%
buildtest/buildsystem/batch.py	75	17	0	44	7	71.43%
buildtest/buildsystem/builders.py	107	24	0	60	8	77.25%
buildtest/buildsystem/compilerbuilder.py	193	36	0	52	10	77.14%
buildtest/buildsystem/parser.py	51	2	4	18	2	94.20%
buildtest/buildsystem/scriptbuilder.py	41	3	0	10	3	88.24%
buildtest/config.py	158	72	0	76	10	47.86%
buildtest/defaults.py	17	17	0	0	0	0.00%
buildtest/docs.py	5	2	0	0	0	60.00%
buildtest/exceptions.py	7	2	2	4	0	81.82%
buildtest/executors/base.py	61	4	4	18	2	89.87%
buildtest/executors/cobalt.py	110	93	0	22	0	12.88%
buildtest/executors/local.py	49	3	0	10	4	88.14%
buildtest/executors/lsf.py	103	85	0	16	0	15.13%
buildtest/executors/pbs.py	96	81	0	14	0	13.64%
buildtest/executors/setup.py	108	60	4	60	8	35.71%
buildtest/executors/slurm.py	110	93	0	28	0	12.32%
buildtest/log.py	19	7	0	0	0	63.16%
buildtest/menu/__init__.py	29	16	113	10	0	33.33%
buildtest/menu/build.py	421	117	0	208	22	70.59%
buildtest/menu/buildspec.py	356	46	0	188	22	83.82%
buildtest/menu/compilers.py	107	60	8	50	3	38.22%
buildtest/menu/config.py	73	4	0	12	1	94.12%
buildtest/menu/inspect.py	63	2	0	46	3	95.41%
buildtest/menu/report.py	193	16	0	114	14	89.58%
buildtest/menu/schema.py	26	0	0	16	2	95.24%
buildtest/schemas/defaults.py	32	0	0	0	0	100.00%
buildtest/schemas/utils.py	26	0	0	8	0	100.00%
buildtest/system.py	155	70	0	38	11	50.78%
buildtest/utils/command.py	68	3	0	20	5	90.91%
buildtest/utils/file.py	68	11	0	30	0	88.78%
buildtest/utils/shell.py	30	0	4	8	0	100.00%
buildtest/utils/timer.py	15	9	0	4	0	31.58%
buildtest/utils/tools.py	19	2	2	12	2	80.65%
Total	3179	1005	145	1246	147	66.19%

For more details on coverage please refer to [coverage documentation](#).

Tox

buildtest provides a [tox.ini](#) configuration to allow user to test regression test in isolated virtual environment. To get started install tox:

```
pip install tox
```

Refer to [tox documentation](#) for more details. To run tox for all environment you can run:

```
tox
```

If your system has one python instance let's say python 3.7 you can test for python 3.7 environment by running `tox -e py37`.

Contributing to Schemas

Schema Docs

Schema Documentation are hosted on branch [gh-pages](#) which is hosted via GitHub Pages at <https://buildtesters.github.io/buildtest/>.

There is an automated workflow [jsonschema2md](#) which publishes schemas, documentation and examples. If you want to edit top-level page [README.md](#) please send a pull-request to *gh-pages* branch.

Adding a new schema

If you want to add a new schema to buildtest you need to do the following:

1. Add schema file in [buildtest/schemas](#) and schema file must end in **.schema.json**. If it's a sub-schema it must in format `<name>-<version>.schema.json`. For example a schema name `script-v2.0.schema.json` will be sub-schema script and version 2.0.
2. Their should be a folder that corresponds to name of schema in [examples](#) directory.
3. There should be a list of invalid and valid examples for schema.
4. There should be regression testfile in [schema_tests](#) to test the schema.

Be sure to update properties and take account for:

- a property being required or not
- Make use of *additionalProperties: false* when defining properties so that additional keys in properties are not passed in.
- requirements for the values provided (types, lengths, etc.)
- If you need help, see [Resources](#) or reach out to someone in Slack.

Running Schema Tests

The schema tests are found in folder `tests/schema_tests` which has regression test for each schema. The purpose for schema test is to ensure Buildsspecs are written according to specification outlined in schemas. Furthermore, we have edge cases to test invalid Buildspec recipes to ensure schemas are working as expected.

To run all schema test you can run via marker:

```
pytest -v -m schema
```

JSON Definitions

We store all JSON definitions in `definitions.schema.json` which are fields need to be reused in other schemas. A JSON definition is defined under `definitions` field, in this example we define a definition anchor **list_of_strings** that declares an array of string:

```
{
  "definitions": {
    "list_of_strings": {
      "type": "array",
      "uniqueItems": true,
      "minItems": 1,
      "items": {"type": "string"}
    },
  }
}
```

A definition anchor can be referenced using **\$ref** keyword. In example below we declare a definitions **string_or_list** that uses **\$ref** that points to anchor `list_of_strings`:

```
"string_or_list": {
  "oneOf": [
    {"type": "string"},
    {"$ref": "#/definitions/list_of_strings"}
  ]
},
```

For example the `tags` field is defined in `definitions.schema.json` that references definition `string_or_list`:

```
"tags": {
  "description": "Classify tests using a tag name, this can be used for categorizing_
↪ test and building tests using ``--tags`` option",
  "$ref": "#/definitions/string_or_list"
},
```

The `tags` field is used in other schemas like `compiler-v1.0.schema.json` and `script-v1.0.schema.json`. In this example we declare **tags** field and reference tags anchor from `definitions.schema.json`:

```
"tags": {
  "$ref": "definitions.schema.json#/definitions/tags"
}
```

It's worth noting each schema must have a **\$id** in order for JSON to resolve references (**\$ref**). For example the definitions schema has the following id:

```
"$id": "definitions.schema.json"
```

It's recommended each schema has a **\$schema**, **\$title**, **description** field for each schema. Currently, we support JSON Schema Draft7 so our schema field must be set to the following:

```
"$schema": "http://json-schema.org/draft-07/schema#",
```

Resources

The following sites (along with the files here) can be useful to help with your development of a schema.

- json-schema.org
- [json schema readthedocs](#)

If you have issues with writing json schema please join the [JSON-SCHEMA Slack Channel](#)

Maintainer Guide

This is a guide for buildtest maintainers

Incoming Pull Request

These are just a few points to consider when dealing with incoming pull requests

1. Any incoming Pull Request should be assigned to one or more maintainers for review.
2. Upon approval, the PR should be **Create a merge commit** or **Squash and merge** depending on your preference. To preserve commit history please use **Create a merge commit** though sometimes it can be useful to do Squash commit. For more details on merge request see <https://docs.github.com/en/github/collaborating-with-pull-requests/incorporating-changes-from-a-pull-request/merging-a-pull-request>
3. Maintainers can request user to put meaningful commit if author has not provided a meaningful message (i.e `git commit --amend`)
4. Maintainers are requested that committer name and email is from a valid Github account. If not please request the committer to fix the author name and email.
5. All incoming PRs should be pushed to [devel](#) branch, if you see any PR sent to any other branch please inform code owner to fix it

Release Process

Every buildtest release will be tagged with a version number using format **X.Y.Z**. Every release will have a tag that corresponds to a release such as `v1.2.3`. Git tags should be pushed to upstream by **release manager** only. The process for pushing git tags can be described in the following article: [Git Basics - Tagging](#)

We will create annotated tags as follows:

```
git tag -a v1.2.3 -m "buildtest version 1.2.3"
```

Once tag is created you can view the tag details by running either:

```
git tag
git show v1.2.3
```

We have created the tag locally, next we must push the tag to the upstream repo by doing the following:

```
git push origin v.1.2.3
```

Every release must have a release note that is maintained in file [CHANGELOG.rst](#)

Under buildtest [releases](#) a new release can be created that corresponds to the git tag. In the release summary, just direct with a message stating **refer to [CHANGELOG.rst](#) for more details**

Once the release is published, make sure to open a pull request from `devel` → `master` and **Rebase and Merge** to master branch. If there are conflicts during merge for any reason, then simply remove `master` and create a master branch from `devel`.

Default Branch

The default branch is `devel` and this should be [protected branch](#).

Branch Settings

All maintainers are encouraged to view branch [settings](#) for `devel` and `master`. The master and devel branches should be protected branches and **devel** branch should be set as the default branch. Shown below is the expected configuration.

The screenshot shows the GitHub repository settings page for `buildtesters/buildtest`. The left sidebar contains a list of settings categories: Options, Manage access, Security & analysis, **Branches** (highlighted), Webhooks, Notifications, Integrations, Deploy keys, Actions, Environments, Secrets, Pages, and Moderation settings. The main content area is titled 'Default branch' and 'Branch protection rules'.

Default branch: The default branch is set to `devel`. A note states: 'The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.'

Branch protection rules: There are two rules defined:

- devel:** Currently applies to 1 branch. Buttons: Edit, Delete.
- master:** Currently applies to 1 branch. Buttons: Edit, Delete.

At the bottom of the rules section are 'Previous' and 'Next' navigation buttons.

If something is not correct please consult with the maintainers.

Merge Settings

We have enabled all commit types i.e (merge commits, squash merging, rebase merging) for merging Pull Request. Shown below is the recommended configuration, if you see a deviation please inform the maintainers.

Merge button

When merging pull requests, you can allow any combination of merge commits, squashing, or rebasing. At least one option must be enabled. If you have linear history requirement enabled on any protected branch, you must enable squashing or rebasing.

- | |
|------------------------------------------------------------------------------------------------------------------------------------------------------|
| <input checked="" type="checkbox"/> Allow merge commits
Add all commits from the head branch to the base branch with a merge commit. |
| <input checked="" type="checkbox"/> Allow squash merging
Combine all commits from the head branch into a single commit in the base branch. |
| <input checked="" type="checkbox"/> Allow rebase merging
Add all commits from the head branch onto the base branch individually. |

If you notice a deviation, please consult with the maintainers.

Google Analytics

The buildtest site is tracked via Google Analytics, if you are interested in get access contact **Shahzeb Siddiqui** @shahzebsiddiqui

Read The Docs Access

buildtest project for readthedocs can be found at <https://readthedocs.org/projects/buildtest/>. If you need to administer project configuration, please contact **Shahzeb Siddiqui** @shahzebsiddiqui to gain access.

Slack Admin Access

If you need admin access to Slack Channel please contact **Shahzeb Siddiqui** @shahzebsiddiqui. The slack admin link is <https://hpcbuildtest.slack.com/admin>

New Maintainers Checklist

Onboarding Email

This guide is to help onboard new maintainers into the buildtest project. To get started send an invitation email as follows:

We are pleased to invite you to the buildtest project **and** become a buildtesters (a.k.a buildtest maintainer). We understand your time **is** valuable; therefore we request a minimal effort of **2-3hrs** per week towards buildtest.

As a buildtesters, you will be working on the following:

(continues on next page)

(continued from previous page)

- * Monitor **and** triage issues
- * Assist user **in** slack channel (*#general*)
- * Update documentation
- * Review **or** triage Pull Request
- * Issue new pull request
- * Troubleshoot build errors **in** regression test **or** CI checks

As a buildtesters you may be granted elevated privilege to the following services: GitHub, ReadTheDocs, Slack, **and** Google Analytics. As a buildtesters, you agree to be accessible on Slack **as** our primary communication channel.

If you agree to these terms, you will be assigned to work **with** another buildtest maintainer **in** your first two weeks. Once you are confident **in** your duties, we will let you work independently at your own pace, should you need help please contact one of the buildtesters.

Please review the contributing guide: <https://buildtest.readthedocs.io/en/devel/contributing.html>

if you are unsure about your responsibilities **as** a buildtesters.

If you agree to these terms **and** conditions, please reply "I **CONFIRM**".

Thanks,
buildtest

Onboarding Checklist

- Please make sure the maintainer has a GitHub account if not please create an account at <https://github.com/join>.
- Ensure user has setup two-factor authentication (2FA) with GitHub.
- Invite member to buildtesters organization.
- Add member to buildtest repository with **Role: Maintain**.
- Invite member to join slack channel and preferably install Slack on your workstation and phone. Please follow instructions to download slack for **Windows**, **Mac**, or **Android**. Slack is available on Apple Store and Google Play Store.
- Once member is added to Slack, ensure member has the appropriate account type. Generally you will want member to be a **Workspace Admin** for more details see [Slack Roles & Permissions](#).
- Ensure member has an account at ReadTheDocs if not please request member to create an account at <https://readthedocs.org/accounts/signup/>. Once member has an account please add member to buildtest readthedocs project at <https://readthedocs.org/dashboard/buildtest/users/>. This will ensure user has ability to access readthedocs platform when troubleshooting build errors related to documentation.

3.13 API Reference

This page contains auto-generated API reference documentation¹.

3.13.1 buildtest

Subpackages

`buildtest.buildsystem`

Submodules

`buildtest.buildsystem.base`

BuilderBase class is an abstract class that defines common functions for any types of builders. Each type schema (script, compiler, spack) is implemented as separate Builder which extends BuilderBase class.

Module Contents

Classes

<i>BuilderBase</i>	The BuilderBase is an abstract class that implements common functions used for building and running test.
--------------------	-----------------------------------------------------------------------------------------------------------

class `buildtest.buildsystem.base.BuilderBase(name, recipe, buildspect, executor, buildexecutor, testdir)`
 Bases: `abc.ABC`

The BuilderBase is an abstract class that implements common functions used for building and running test.

buildtest will create a builder object which resembles a test. The builder will contains metadata that is unique to each builder that is captured in the report file upon completion of test.

name

name of test

Type `str`

metadata

stores metadata for test

Type `dict`

recipe

stores loaded recipe for test section

Type `dict`

executor

stores name of executor used for running builder

Type `str`

¹ Created with sphinx-autoapi

job

Stores Job Data for batch job

Type `buildtest.executors.job.Job`

state

Determines state of builder

Type `bool`

buildspec

Store full path to buildspec file

Type `str`

testdir

Root of test directory where test will live.

Type `str`

buildexecutor (obj

`buildtest.executors.setup.BuildExecutor`): instance of BuildExecutor class used for accessing executors

shebang

used for writing shebang line in test. This can vary based on shell type or if `shebang` property is specified

Type `str`

shell (obj

`buildtest.utils.shell.Shell`): An instance of Shell class used for detecting shell type

The BuilderBase provides common functions for any builder. The builder is an instance of BuilderBase. The initializer method will setup the builder attributes based on input test by `name` parameter.

Parameters

- **name** (`str`) – Name of test in buildspec recipe
- **recipe** (`str`) – The loaded test section from the buildspec file
- **buildspec** (`str`) – Full path to buildspec file
- **buildexecutor** (`buildtest.executors.setup.BuildExecutor`) – An instance of BuildExecutor class used for accessing executors
- **testdir** (`str`) – Test directory where tests are written. Must be full path on filesystem.

`_set_metadata_values(self)`

This method sets `self.metadata` that contains metadata for each builder object.

`_generate_unique_id(self)`

Generate a unique build id using `uuid.uuid4()`.

Returns unique test id for the builder

Return type `str`

`get_test_extension(self)`

Return the test extension, which depends on the shell type. By default we return `sh` file extension for all shells except for `csh` which will return “csh” extension.

Returns `str`: Returns test extension name for generated test.

`_is_local_executor(self)`

Return True if current builder executor type is LocalExecutor otherwise returns False.

Returns returns True if builder is using executor type LocalExecutor otherwise returns False

Return type `bool`

is_batch_job(*self*)

Return True/False if builder.job attribute is of type Job instance if not returns False. This method indicates if builder has a job submitted to queue

start(*self*)

Keep internal timer for test using class `buildtest.utils.timer.Timer`. This method will start the timer for builder which is invoked upon running test.

stop(*self*)

Stop internal timer for builder.

build(*self*)

This method is responsible for invoking setup, creating test directory and writing test. This method is called from an instance object of this class that does `builder.build()`.

retry(*self*, *retry*)

run(*self*)

Run the test and record the starttime and start timer. We also return the instance object of type BuildTestCommand which is used by Executors for processing output and error

Returns

If success, the return type is an object of type `buildtest.utils.command.BuildTestCommand`

If there is a failure (non-zero) returncode we retry test and if it doesn't pass we raise exception of `buildtest.exceptions.RuntimeFailure`

starttime(*self*)

This method will record the starttime when job starts execution by using `datetime.datetime.now()`

endtime(*self*)

This method is called upon termination of job, we get current time using `datetime.datetime.now()` and calculate runtime of job

runtime(*self*)

Calculate runtime of job by calculating delta between endtime and starttime. The unit of measure is seconds.

get_runtime(*self*)

Return runtime of test

success(*self*)

This method is invoked to indicate that builder job is complete after polling job.

failure(*self*)

This method indicates that builder job is not complete after polling job either job was cancelled by scheduler or job failed to run.

is_complete(*self*)

If builder completes execution of test this method will return True otherwise returns False. A builder could fail due to job cancellation, failure to submit job or raise exception during the run phase. In those case, this method will return False.

is_failure(*self*)

Return True if builder fails to run test.

is_unknown(*self*)

Returns True if builder state is unknown which is the state if job is still running

run_command(self)

Command used to run the build script. buildtest will change into the stage directory (self.stage_dir) before running the test.

copy_stage_files(self)

Copy output and error file into test root directory.

_build_setup(self)

This method is the setup operation to get ready to build test which includes the following:

1. Creating Test directory and stage directory
2. Resolve full path to generated test script and build script
3. Copy all files from buildspect directory to stage directory

_emit_command(self)

This method will return a shell command used to invoke the script that is used for tests that use local executors

Returns

a list to show generated command used to run test.

Test can be run without any argument with path to script: `/path/to/script.sh` Test can be run with shell name followed by path to script: `bash /path/to/script.sh` Test can be run with shell name, shell options and path to script: `bash -x /path/to/script.sh`

Return type `list`**_default_test_variables(self)**

Return a list of lines inserted in build script that define buildtest specific variables that can be referenced when writing tests. The buildtest variables all start with BUILDTEST_*

_write_build_script(self)

This method will write the content of build script that is run for when invoking the builder run method. Upon creating file we set permission of builder script to 755 so test can be run.

_write_test(self)

This method is responsible for invoking `generate_script` that formulates content of testscript which is implemented in each subclass. Next we write content to file and apply 755 permission on script so it has executable permission.

sched_init(self)

This method will resolve scheduler fields: 'sbatch', 'pbs', 'bsub', 'cobalt'

get_slurm_directives(self)

Get #SBATCH lines based on sbatch property by calling `buildtest.buildsystem.batch.SlurmBatchScript`

get_lsf_directives(self)

Get #BSUB lines based on bsub property by calling `buildtest.buildsystem.batch.LSFBatchScript`

get_pbs_directives(self)

Get #PBS lines based on pbs property by calling `buildtest.buildsystem.batch.PBSBatchScript`

get_cobalt_directives(self)

Get #COBALT lines based on cobalt property by calling `buildtest.buildsystem.batch.CobaltBatchScript`

get_job_directives(self)

This method returns a list of lines containing the scheduler directives

`_get_burst_buffer(self, burstbuffer)`

Get Burst Buffer directives (**#BB**) lines specified by BB property

Parameters **burstbuffer** (*str*) – Burst Buffer configuration specified by BB property in buildspec

Returns List of string values containing containing #BB directives written in test

Return type *list*

`_get_data_warp(self, datawarp)`

Get Cray Data Warp directives (**#DW**) lines specified by DW property.

Parameters **datawarp** (*str*) – Data Warp configuration specified by DW property in buildspec

Returns List of string values containing containing #DW directives written in test

Return type *list*

`_set_execute_perm(self, fname)`

Set permission to 755 for a given file. The filepath must be an absolute path to file

`_get_environment(self, env)`

Retrieve a list of environment variables defined in buildspec and return them as list with the shell equivalent command

Parameters **env** (*dict*) – list of environment variables defined by env property in buildspec

`_get_variables(self, variables)`

Retrieve a list of variables defined in buildspec and return them as list with the shell equivalent command.

Parameters **variables** (*dict*) – list of variable defined by vars property in buildspec

`add_metrics(self)`

This method will update the metrics field stored in `self.metadata['metrics']`. The `metrics` property can be defined in the buildspec to assign value to a metrics name based on regular expression, environment or variable assignment.

`output(self)`

Return output content

`error(self)`

Return error content

`abstract generate_script(self)`

Build the testscript content implemented in each subclass

`post_run_steps(self)`

This method is called after test is complete. This method will copy files from stage directory such as output, error and test script. We will check state of test and mark job is complete.

`_check_regex(self)`

This method conducts a regular expression check using `re.search` with regular expression defined in Buildspec. User must specify an output stream (stdout, stderr) to select when performing regex. In buildtest, this would read the `.out` or `.err` file based on stream and run the regular expression to see if there is a match. This method will return a boolean `True` indicates there is a match otherwise `False` if `regex` object not defined or `re.search` doesn't find a match.

Returns Returns `True` if their is a regex match otherwise returns `False`.

Return type *bool*

`_returncode_check(self)`

Check status check of `returncode` field if specified in status property.

_check_runtime(*self*)

This method will return a boolean (True/False) based on runtime specified in builds spec and check with test runtime. User can specify both *min* and *max*, or just specify *min* or *max*.

check_test_state(*self*)

This method is responsible for detecting state of test (PASS/FAIL) based on returncode or regular expression.

__str__(*self*)

Return str(*self*).

__repr__(*self*)

Return repr(*self*).

buildtest.buildsystem.batch**Module Contents****Classes**

<i>BatchScript</i>	Base class used for generating Batch directives for Schedulers
<i>LSFBatchScript</i>	This class is responsible for building LSF batch script by taking bsub property and converting them into #BSUB directives
<i>SlurmBatchScript</i>	This class is responsible for building Slurm batch script by taking sbatch property and converting them into #SBATCH directives
<i>CobaltBatchScript</i>	This class is responsible for building Cobalt batch script by taking cobalt property and converting them into #COBALT directives
<i>PBSBatchScript</i>	This class is responsible for building PBS batch script by taking pbs property and converting them into #PBS directives

class buildtest.buildsystem.batch.BatchScript

Base class used for generating Batch directives for Schedulers

get_headers(*self*)**class buildtest.buildsystem.batch.LSFBatchScript(*bsub*)**

Bases: *BatchScript*

This class is responsible for building LSF batch script by taking bsub property and converting them into **#BSUB** directives

This method will return a list of **#BSUB** directives used in job script

Parameters *bsub* (*list*) – List of string items specified by bsub property in builds spec used for specified **#BSUB** directive

Returns A list of **#BSUB** directive that will be inserted for LSF Job Script

Return type *list*

class buildtest.buildsystem.batch.SlurmBatchScript(*sbatch*)

Bases: *BatchScript*

This class is responsible for building Slurm batch script by taking `sbatch` property and converting them into `#SBATCH` directives

This method will return a list of `#Slurm` directives used in job script

Args: `sbatch` (list): List of string items specified by `sbatch` property in builds spec used for specified `#SBATCH` directive

Returns A list of `#SBATCH` directive that will be inserted for Slurm Job Script

Return type `list`

class `buildtest.buildsystem.batch.CobaltBatchScript(cobalt)`

Bases: `BatchScript`

This class is responsible for building Cobalt batch script by taking `cobalt` property and converting them into `#COBALT` directives

This method will return a list of `#Cobalt` directives used in job script

Args: `cobalt` (list): List of string items specified by `cobalt` property in builds spec used for specified `#COBALT` directive

Returns A list of `#COBALT` directive that will be inserted for Cobalt Job Script

Return type `list`

class `buildtest.buildsystem.batch.PBSBatchScript(pbs)`

Bases: `BatchScript`

This class is responsible for building PBS batch script by taking `pbs` property and converting them into `#PBS` directives

This method will return a list of `#PBS` directives used in job script

Parameters `pbs` (`list`) – List of string items specified by `pbs` property in builds spec used for specified `#PBS` directive

Returns A list of `#PBS` directive that will be inserted for PBS Job Script

Return type `list`

`buildtest.buildsystem.builders`

This module implements the Builder class that is responsible for getting builders from a builds spec file. The Builder class is invoked once builds spec file has parsed validation via `buildtest.buildsystem.parser.BuildspecParser`.

Module Contents

Classes

<code>Builder</code>	The Builder class creates builder objects based on parsed builds specs.
----------------------	-------------------------------------------------------------------------

```
class buildtest.buildsystem.builders.Builder(bp, buildexecutor, filters, testdir, configuration,
                                             buildtest_system=None, rebuild=1)
```

The Builder class creates builder objects based on parsed buildsspecs.

The builder class is created based on the 'type' field in the test. If test contains `type: script` we will create builder by calling `buildtest.buildsystem.scriptbuilder.ScriptBuilder`. Likewise for `type: compiler` and `type: spack` we will call `buildtest.buildsystem.compilerbuilder.CompilerBuilder` and `buildtest.buildsystem.spack.SpackBuilder`.

Based on a loaded Buildspeg file, return the correct builder for each based on the type. Each type is associated with a known Builder class.

Parameters

- **bp** (`buildtest.buildsystem.parser.BuildspecParser`) – Instance of Buildspec-Parser class
- **buildexecutor** (`buildtest.executors.setup.BuildExecutor`) – Instance of BuildExecutor class
- **filters** (*dict*) – List of filter fields specified via `buildtest build --filter` for filtering tests
- **testdir** (*str*) – Test directory where tests will be written which could be specified via `buildtest build --testdir` or configuration file
- **configuration** (`buildtest.config.SiteConfiguration`) – Instance of SiteConfiguration class
- **buildtest_system** (`buildtest.system.BuildTestSystem`, *optional*) – Instance of BuildTestSystem class
- **rebuild** (*int*, *optional*) – Number of rebuild for test. This is specified via `buildtest build --rebuild`. Defaults to 1

```
_generate_builders(self, recipe, name, compiler_name=None)
```

This method is responsible for generating builders by applying regular expression specified by `executor` field in buildspeg with list of executors. If their is a match we generate a builder.

Parameters

- **name** (*str*) – Name of test in buildspeg file
- **recipe** (*dict*) – Loaded test recipe from buildspeg file
- **compiler_name** (*str*, *optional*) – Name of compiler

Returns List of builder objects

```
_build_compilers(self, name, recipe)
```

This method will perform regular expression with 'name' field in compilers section and retrieve one or more compiler that were defined in buildtest configuration. If any compilers were retrieved we return one or more builder objects that call `buildtest.buildsystem.compilerbuilder.CompilerBuilder`

Parameters

- **name** (*str*) – name of test
- **recipe** (*dict*) – Loaded test recipe from buildspeg

```
_skip_tests_by_tags(self, recipe, name)
```

This method determines if test should be skipped based on tag names specified in filter field that is specified on command line via `buildtest build --filter tags=<TAGNAME>`

Parameters

- **recipe** (*dict*) – Loaded test recipe from buildspect
- **name** (*str*) – Name of test

Returns False if buildtest build --filter tags is not specified. If specified we return True if tags field is not in test recipe or there is a matching tag.

Return type bool

_skip_tests_by_type(*self, recipe, name*)

This method determines if test should be skipped based on type field specified in filter field that is specified on command line via buildtest build --filter type=<SCHEMATYPE>

Parameters

- **recipe** (*dict*) – Loaded test recipe from buildspect
- **name** (*str*) – Name of test

Returns False if buildtest build --filter type is not specified. If there is a match with input filter and type field in test we return True

Return type bool

_skip_tests_run_only(*self, recipe, name*)

This method will skip tests based on run_only field from buildspect. Checks are performed based on conditionals and if any conditional is not met we skip test.

Parameters

- **recipe** (*dict*) – Loaded test recipe from buildspect
- **name** (*str*) – Name of test

Returns

False if run_only property not specified in buildspect otherwise returns True based on following condition

- True if there is no match with system ‘scheduler’ and one specified in buildspect
- True if there is no match with user specified by ‘user’ property and one detected by system using `os.getenv("USER")`
- True if there is no match with specified ‘platform’ property and one detected by system platform
- True if there is no match with specified ‘linux_distro’ property and one detected by system

Return type bool

get_builders(*self*)

Return a list of builder objects

get_test_names(*self*)

Return the list of test names for the loaded Buildspect recipe

buildtest.buildsystem.compilerbuilder

Module Contents

Classes

CompilerBuilder

This is a subclass of BuilderBase used for building test that uses `type: compiler` in the builds spec.

class buildtest.buildsystem.compilerbuilder.**CompilerBuilder**(*name, recipe, builds spec, buildexecutor, executor, configuration, compiler=None, testdir=None*)

Bases: *buildtest.buildsystem.base.BuilderBase*

This is a subclass of BuilderBase used for building test that uses `type: compiler` in the builds spec.

The BuilderBase provides common functions for any builder. The builder is an instance of BuilderBase. The initializer method will setup the builder attributes based on input test by `name` parameter.

Parameters

- **name** (*str*) – Name of test in builds spec recipe
- **recipe** (*str*) – The loaded test section from the builds spec file
- **builds spec** (*str*) – Full path to builds spec file
- **buildexecutor** (*buildtest.executors.setup.BuildExecutor*) – An instance of BuildExecutor class used for accessing executors
- **testdir** (*str*) – Test directory where tests are written. Must be full path on filesystem.

type = `compiler`

lang_ext_table

cc

cxx

fc

ldflags

cflags

cxxflags

fflags

cppflags

setup(*self*)

The setup method is responsible for process compiler section, getting modules `pre_build`, `post_build`, `pre_run`, `post_run` section and generate compilation and run command. This method invokes other methods and set values in class variables. This method is called by `self.generate_script` method.

generate_script(*self*)

This method is responsible for generating test script for compiler schema. The method `generate_script` is implemented in each subclass because implementation on test generation differs across schema types.

This method will add the lines into list which comprise content of test. The method will return a list containing lines of test script.

_resolve_source(*self*)

This method resolves full path to source file, it checks for absolute path first before checking relative path that is relative to Buildspeg recipe.

_detect_lang(*self*, *sourcefile*)

This method will return the Programming Language based by looking up file extension of source file.

_get_modules(*self*, *modules*)

Return a list of module command as a list of instructions based on module property.

param modules 'module' property specified in buildspeg used for loading/swapping
modules

type modules object

_compile_cmd(*self*)

This method generates the compilation line and returns the output as a list. The compilation line depends on the the language detected that is stored in variable `self.lang`.

_run_cmd(*self*)

This method builds the run command which refers to how to run the generated binary after compilation.

_process_compiler_config(*self*)

This method is responsible for setting cc, fc, cxx class variables based on compiler selection. The order of precedence is `config`, `default`, then buildtest setting. Compiler settings in 'config' takes highest precedence, this overrides any configuration in 'default'. Finally we resort to compiler configuration in buildtest setting if none defined. This method is responsible for setting cc, fc, cxx, cflags, cxxflags, fflags, ldflags, and cppflags.

set_cc(*self*, *cc*)

set_cxx(*self*, *cxx*)

set_fc(*self*, *fc*)

set_cflags(*self*, *cflags*)

set_fflags(*self*, *fflags*)

set_cxxflags(*self*, *cxxflags*)

set_cppflags(*self*, *cppflags*)

set_ldflags(*self*, *ldflags*)

get_cc(*self*)

get_cxx(*self*)

get_fc(*self*)

get_cflags(*self*)

get_cxxflags(*self*)

get_fflags(*self*)

get_cppflags(*self*)

get_ldflags(*self*)

get_path(*self*)

This method returns the full path for C, C++, Fortran compilers

buildtest.buildsystem.parser

BuildspecParser is will validate a Buildspec file with the global schema and each test will be validated with one of the subschemnas based on the type field. The BuilderBase is the base class for all builders that expose functions to run builds.

Module Contents

Classes

<i>BuildspecParser</i>	A BuildspecParser is responsible for validating a Buildspec file with JSON Schema.
------------------------	------------------------------------------------------------------------------------

class buildtest.buildsystem.parser.**BuildspecParser**(*buildspec*, *buildexecutor*)

A BuildspecParser is responsible for validating a Buildspec file with JSON Schema.

The type (e.g., script) and version are derived from reading in the file, and then matching to a Buildspec schema.

The schemas are located in folder `buildtest/schemas`, we load the schema dictionary and validate each buildspec with global schema and a sub-schema based on the `type` field. If the schema fails validation check, then we stop immediately.

The init method will run some checks against buildspec before loading buildspec. buildtest will validate the entire buildspec with `global.schema.json` and validate each test section with the designated type schema. For instance of test includes `type: script` we will validate the test with schema `script-v1.0.schema.json`.

If there is any error during the init method, an exception will be raised.

Parameters

- **buildspec** (*str*) – Full path to buildspec file
- **buildexecutor** (`buildtest.executors.setup.BuildExecutor`) – Instance object of class BuildExecutor used for accessing executors which is created based on configuration file

Raises *BuildTestError* – Raise exception if there is issue with buildexecutor, or buildspec is not resolved to file path. If buildspec is a directory path we raise an exception

__str__(*self*)
Return str(self).

__repr__(*self*)
Return repr(self).

_check_schema_type(*self*, *test*)
Check type field is a valid sub-schema and verify type + version will resolve to a schema file.

Parameters **test** (*str*) – Name of test in buildspecs property of buildspec file

Raises *BuildspecError* – If there is no match with type property in test with available schema types

_check_executor(*self*, *test*)
This method checks if executor property is not None and executor value is found in list of available executors.

Parameters **test** (*str*) – Name of test in buildspecs property of buildspec file

Raises *BuildspecError* – If there is no match with `executor` property in test with list of available executors

`_validate(self)`

This method will validate the entire buildspec file with global schema and each test section with a sub-schema. The global validation ensures that the overall structure of the file is sound for further parsing. We load in the `global.schema.json` for this purpose.

A buildspec is composed of one or more tests, each section is validated with a sub-schema. The `type` field is used for sub-schema lookup from schema library. Finally we validate loaded recipe with sub-schema.

buildtest.buildsystem.scriptbuilder

Module Contents

Classes

<i>ScriptBuilder</i>	This is a subclass of <code>BuilderBase</code> used for building test that uses <code>type: script</code> in the buildspec.
----------------------	-----------------------------------------------------------------------------------------------------------------------------

class `buildtest.buildsystem.scriptbuilder.ScriptBuilder`(*name, recipe, buildspec, executor, buildexecutor, testdir*)

Bases: `buildtest.buildsystem.base.BuilderBase`

This is a subclass of `BuilderBase` used for building test that uses `type: script` in the buildspec.

The `BuilderBase` provides common functions for any builder. The builder is an instance of `BuilderBase`. The initializer method will setup the builder attributes based on input test by `name` parameter.

Parameters

- **name** (*str*) – Name of test in buildspec recipe
- **recipe** (*str*) – The loaded test section from the buildspec file
- **buildspec** (*str*) – Full path to buildspec file
- **buildexecutor** (`buildtest.executors.setup.BuildExecutor`) – An instance of `BuildExecutor` class used for accessing executors
- **testdir** (*str*) – Test directory where tests are written. Must be full path on filesystem.

type = `script`

`write_python_script(self)`

This method is used for writing python script when `shell: python` is set. The content from `run` section is added into a python script. The file is written to `run` directory and we simply invoke python script by running `python script.py`

`generate_script(self)`

This method builds the content of the test script which will return a list of shell commands that will be written to file.

A typical test will contain: shebang line, job directives, environment variables and variable declaration, and content of `run` property. For `shell: python` we write a python script and return immediately. The variables, environment section are not applicable for python scripts

Returns List of shell commands that will be written to file

buildtest.buildsystem.spack

This method defines the Spack buildsystem for the spack package manager (<https://spack.readthedocs.io/en/latest/>) by generating scripts that will do various spack operation. The SpackBuilder class will generate a test script using the schema definition 'spack-v1.0.schema.json' that defines how buildsspecs are written.

Module Contents

Classes

<i>SpackBuilder</i>	This is a subclass of BuilderBase used for building test that uses <code>type: spack</code> in the buildspec.
---------------------	---------------------------------------------------------------------------------------------------------------

class buildtest.buildsystem.spack.**SpackBuilder**(*name, recipe, buildspec, buildexecutor, executor, testdir=None*)

Bases: *buildtest.buildsystem.base.BuilderBase*

This is a subclass of BuilderBase used for building test that uses `type: spack` in the buildspec.

The BuilderBase provides common functions for any builder. The builder is an instance of BuilderBase. The initializer method will setup the builder attributes based on input test by name parameter.

Parameters

- **name** (*str*) – Name of test in buildspec recipe
- **recipe** (*str*) – The loaded test section from the buildspec file
- **buildspec** (*str*) – Full path to buildspec file
- **buildexecutor** (*buildtest.executors.setup.BuildExecutor*) – An instance of BuildExecutor class used for accessing executors
- **testdir** (*str*) – Test directory where tests are written. Must be full path on filesystem.

type = `spack`

generate_script(*self*)

Method responsible for generating the content of test script for spack buildsystem

_resolve_spack_root(*self, path, verify_spack=True*)

Given a path find the startup spack setup script to source.

Parameters

- **path** (*str*) – Full path to root of spack directory
- **verify_spack** (*bool, optional*) – Check for existence of spack setup script `$SPACK_ROOT/share/spack/setup-env.sh` before sourcing file. By default this check is enabled but can be disabled to allow test to run even if spack doesn't exist on filesystem.

Raises *BuildTestError* – Raise exception if root of spack doesn't exist or we are unable to resolve path to setup script `$SPACK_ROOT/share/spack/setup-env.sh`

_spack_environment(*self, spack_env*)

This method is responsible for creating a spack environment, activate an existing spack environment, create a spack environment from a directory and a manifest file (spack.yaml, spack.lock)

Parameters **spack_env** (*dict*) – Contains property env from buildspec dictionary

buildtest.cli

buildtest cli: include functions to build, get test configurations, and interact with a global configuration for buildtest.

Submodules

buildtest.cli.build

This module contains all the methods related to “buildtest build” which is used for building test scripts from a Builds spec

Module Contents

Classes

<i>BuildTest</i>	This class is an interface to building tests via buildtest build command.
------------------	----------------------------------------------------------------------------------

Functions

<i>resolve_testdirectory</i> (configuration, test-dir=None)	This method resolves which test directory to select. For example, one
<i>discover_buildspecs</i> (buildspecs=None, exclude_buildspecs=None, executors=None, tags=None)	This method discovers all buildspecs based on --buildspecs , --tags , --executor
<i>print_discovered_buildspecs</i> (buildspec_dict)	This method will print the discovered buildspecs in table format
<i>discover_buildspecs_by_tags</i> (tagnames)	This method discovers buildspecs by tags, using buildtest build --tags option.
<i>discover_buildspecs_by_executor</i> (executors)	This method discovers buildspecs by executor name, using buildtest build --executor
<i>discover_by_buildspecs</i> (buildspec)	Given a buildspec file specified by the user with buildtest build --buildspec ,
<i>print_filters</i> ()	This method will print list of filters fields used by buildtest build --filter . This method is invoked by
<i>update_report</i> (valid_builders, report_file=BUILD_REPORT)	This method will update BUILD_REPORT after every test run performed

Attributes

<i>logger</i>

buildtest.cli.build.logger

buildtest.cli.build.resolve_testdirectory(configuration, testdir=None)

This method resolves which test directory to select. For example, one can specify test directory via

command line `buildtest build --testdir <path>` or path in configuration file. The default is `$HOME/.buildtest/var/tests`

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of SiteConfiguration class which contains content of buildtest configuration file
- **testdir** (`str`, *optional*) – Path to test directory specified via command line `buildtest build --testdir`

Returns Path to test directory

Return type `str`

`buildtest.cli.build.discover_buildspecs(buildspecs=None, exclude_buildspecs=None, executors=None, tags=None)`

This method discovers all buildspecs based on `--buildspecs`, `--tags`, `--executor` and excluding buildspecs (`--exclude`).

Parameters

- **buildspecs** (`list`, *optional*) – List of input buildspecs passed by argument `buildtest build --buildspec`
- **exclude_buildspecs** (`list`, *optional*) – List of excluded buildspecs by argument `buildtest build --exclude`
- **tags** (`list`, *optional*) – List of input tags for discovering buildspecs by argument `buildtest build --tags`
- **executors** (`list`, *optional*) – List of input executors for discovering buildspecs by argument `buildtest build --executor`

Returns A dictionary containing a list of included, excluded, detected buildspecs and buildspecs detected based on tags and executors

Return type `dict`

`buildtest.cli.build.print_discovered_buildspecs(buildspec_dict)`

This method will print the discovered buildspecs in table format

Parameters **buildspec_dict** (`dict`) – A dictionary containing a list of included and excluded buildspecs and breakdown of buildspecs by tags and executors

`buildtest.cli.build.discover_buildspecs_by_tags(tagnames)`

This method discovers buildspecs by tags, using `buildtest build --tags` option. This method will read `BUILDSPEC_CACHE_FILE` and search for `tags` key in buildspec recipe and match with input tag. The input tags are a list of tagnames to search in buildspec with the `tags` property in buildspec. The return is a list of buildspec files to process.

Parameters **tagnames** (`list`) – List of input tags from command line argument `buildtest build --tags <tags>`

Returns first argument is a list of buildspecs discovered for all tag names. The second argument is dictionary breakdown of buildspecs by each tag name

Return type `list, dict`

`buildtest.cli.build.discover_buildspecs_by_executor(executors)`

This method discovers buildspecs by executor name, using `buildtest build --executor` command. This method will read `BUILDSPEC_CACHE_FILE` and search for `executor` property in buildspec and match with input executor name. The return is a list of matching buildspec with executor name to process.

Parameters `executors` (*list*) – List of input executor name from command line argument
`buildtest build --executor <name>`

Returns first argument is a list of buildsspecs discovered for all executors. The second argument is dictionary breakdown of buildsspecs by each executor name

Return type *list, dict*

`buildtest.cli.build.discover_by_buildspecs(buildspec)`

Given a buildspec file specified by the user with `buildtest build --buildspec`, discover one or more files and return a list for buildtest to process. This method is called once per argument of `--buildspec` or `--exclude` option. If its a directory path we recursively find all buildsspecs with with `.yaml` extension. If filepath doesn't exist or file extension is not `.yaml` we return None and capture error in log.

```
# file path
buildtest build --buildspec tutorials/hello.sh.yaml

# directory path
buildtest build --buildspec tutorials

# invalid file path returns None
buildtest build -b /xyz.yaml

# invalid file extension
buildtest build -b README.md
```

Parameters `buildspec` (*str*) – Full path to buildspec based on argument `buildtest build --buildspec`

Returns List of resolved buildsspecs.

Return type *list*

`buildtest.cli.build.print_filters()`

This method will print list of filters fields used by `buildtest build --filter`. This method is invoked by running `buildtest build --helpfilter`.

class `buildtest.cli.build.BuildTest(configuration=None, buildsspecs=None, exclude_buildspecs=None, tags=None, executors=None, testdir=None, stage=None, filter_buildspecs=None, rebuild=None, buildtest_system=None, report_file=None, max_pend_time=None, poll_interval=None, keep_stage_dir=None, retry=None, helpfilter=None)`

This class is an interface to building tests via `buildtest build` command.

The initializer method is responsible for checking input arguments for type check, if any argument fails type check we raise an error. If all arguments pass we assign the values and proceed with building the test.

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`, *optional*) – Loaded configuration content which is an instance of `SiteConfiguration`
- **buildspecs** (*list*, *optional*) – list of buildsspecs from command line `buildtest build --buildspec`
- **exclude_buildspecs** (*list*, *optional*) – list of excluded buildsspecs from command line `buildtest build --exclude`
- **tags** (*list*, *optional*) – list if tags passed from command line `buildtest build --tags`

- **executors** (*list*, *optional*) – list of executors passed from command line buildtest build --executors
- **testdir** (*str*) – Path to test directory where tests are written. This argument can be passed from command line buildtest build --testdir
- **stage** (*str*, *optional*) – Stop build after parse or build stage which can be configured via buildtest build --stage option
- **filter_buildspecs** (*dict*, *optional*) – filters buildspecs and tests based on buildtest build --filter argument which is a key/value dictionary that can filter tests based on **tags**, **type**, and **maintainers**
- **rebuild** (*int*, *optional*) – Rebuild tests X times based on buildtest build --rebuild option.
- **buildtest_system**(*buildtest.system.BuildTestSystem*, *optional*) – Instance of BuildTestSystem class
- **report_file** (*str*, *optional*) – Location to report file where test data will be written upon completion. This can be specified via buildtest build --report command
- **max_pend_time** (*int*, *optional*) – Specify maximum pending time in seconds for batch job until job is cancelled
- **poll_interval** (*int*, *optional*) – Specify poll interval in seconds for polling batch jobs.
- **keep_stage_dir** (*bool*, *optional*) – Keep stage directory after job completion
- **retry** (*int*, *optional*) – Number of retry for failed jobs
- **helpfilter** (*bool*, *optional*) – Display available filter fields for buildtest build --filter command. This argument is set to True if one specifies buildtest build --helpfilter

_validate_filters(*self*)

Check filter fields provided by buildtest build --filter are valid types and supported. Currently supported filter fields are tags, type, maintainers

Raises *BuildTestError* – if input filter field is not valid we raise exception. For type filter we check for value and make sure the schema type is supported

build(*self*)

This method is responsible for discovering buildspecs based on input argument. Then we parse the buildspecs and retrieve builder objects for each test. Each builder object will invoke *buildtest.buildsystem.base.BuilderBase.build()* which will build the test script, and then we run the test and update report.

parse_buildspecs(*self*)

Parse all buildspecs by passing buildspec file to *buildtest.buildsystem.parser.BuildspecParser* class. If buildspec fails validation we skip the buildspec and print all skipped buildspecs. If buildspec passes validation we get all builders by invoking Builder class that is responsible for creating builder objects for each test.

Raises *SystemExit* – If no builders are created after parsing buildspecs

build_phase(*self*)

This method will build all tests by invoking class method build for each builder that generates testscript in the test directory. If no builders are present upon building test we raise exception and terminate immediately

Raises *BuildTestError* – If no builders are present in build phase

run_phase(self)

This method will run all builders with the appropriate executor. The `buildtest.executors.setup.BuildExecutor` class is responsible for orchestrating builder execution to the appropriate executor class. The BuildExecutor contains a list of executors picked up from buildtest configuration. For tests running locally, we get the test metadata and count PASS/FAIL test state which is printed at end in Test Summary. For tests that need batch submission via scheduler, the first stage of run will dispatch job, and state will be unknown. After dispatching all jobs, we will poll jobs until they are complete. The poll section is skipped if all tests are run locally. In poll section we regenerate table with all valid builders and updated test state and returncode and recalculate total pass/fail tests. Any test that failed to run or be dispatched will be skipped during run stage and they will not be recorded in the test report

Returns A list of valid builders after running tests

poll_phase(self, builders)

This method will poll jobs by processing all builders. The `buildtest.executors.poll.PollQueue` is responsible for polling builders at set interval until all jobs are complete. The PollQueue will cancel job if it exceeds `max_pend_time` to ensure jobs are not stuck indefinitely. If job is cancelled by scheduler, we remove this from list of builders that will be returned from this method. This method will return a list of valid builders after polling. If there are no valid builders after polling, the method will return **None**

Parameters **builders** (*list*) – List of builder objects that require polling

Returns List of builder objects that ran to completion without any failure.

_print_build_phase(self, invalid_builders, table)

print build phase table

Parameters

- **invalid_builders** (*list*) –
- **table** (*dict*) – a dict mapping of builders printed in build phase.

_print_run_phase(self, builders)

Print run phase table

Parameters **builders** (*list*) – List of builders to print in run phase

_print_test_summary(self, builders)

Print a summary of total pass and fail test with percentage breakdown.

Parameters **builders** (*list*) – List of builders that ran to completion

_update_build_history(self, builders)

Write a build history file that is stored in `$BUILDTEST_ROOT/var/.history` directory summarizing output of build. The history file is a json file named `build.json` which contains a copy of the build log for troubleshooting. buildtest will create a sub-directory that is incremented such as 0, 1, 2 in `$BUILDTEST_ROOT/var/.history` which is used to differentiate builds.

Shown below is content of the top-level directory for the history directory. There is one subdirectory for each build ID starting with 0

```
bash-3.2$ ls -l $BUILDTEST_ROOT/var/.history
total 0
drwxr-xr-x  4 siddiq90  92503  128 Sep  8 13:50 0
drwxr-xr-x  4 siddiq90  92503  128 Sep  8 13:50 1
```

For every build ID we have a `build.json` and log file for each build.


```

bash-3.2$ ls $BUILDTEST_ROOT/var/.history/{0,1}
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/.history/0:
build.json                buildtest_y3gh46j_.log

/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/.history/1:
build.json                buildtest_alrjdy59.log

```

buildtest.cli.build.update_report(*valid_builders*, *report_file*=*BUILD_REPORT*)

This method will update BUILD_REPORT after every test run performed by buildtest build. If BUILD_REPORT is not created, we will create file and update json file by extracting contents from builder metadata

Parameters

- **valid_builders** (*list*) – List of valid builders that ran to completion
- **report_file** (*str*) – Specify location to report file.

buildtest.cli.buildspec

Module Contents

Classes

<i>BuildspecCache</i>	The initializer method for BuildspecCache class is responsible for loading and finding buildspecs into buildspec cache. First we
-----------------------	----------------------------------------------------------------------------------------------------------------------------------

Functions

<i>show_buildspecs</i> (name, configuration)	This is the entry point for buildtest buildspec show command which will print content of
<i>buildspec_validate</i> (configuration, buildspecs=None, excluded_buildspecs=None, tags=None, executors=None)	Entry point for buildtest buildspec validate. This method is responsible for discovering buildspec
<i>summarize_buildspec_cache</i> (configuration)	Prints summary of buildspec cache which is run via command buildtest buildspec summary
<i>buildspec_find</i> (args, configuration)	Entry point for buildtest buildspec find command

Attributes

logger

`buildtest.cli.buildspec.logger`

class `buildtest.cli.buildspec.BuildspecCache`(*configuration, rebuild=False, filterfields=None, formatfields=None, roots=None, header=None, terse=None*)

The initializer method for BuildspecCache class is responsible for loading and finding buildspecs into buildspec cache. First we resolve paths to directory where buildspecs will be searched. This can be specified via `--roots` option on command line or one can specify directory paths in the configuration file. Next we build the cache that contains metadata for each buildspec that will be written to file. If any filter or format options are specified we check if they are valid and finally display a content of the cache depending on the argument.

This method is called when using `buildtest buildspec find` command.

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`) – Instance of SiteConfiguration class that is loaded buildtest configuration.
- **rebuild** (*bool, optional*) – rebuild the buildspec cache by validating all buildspecs when using `buildtest buildspec find --rebuild`. Defaults to False if `--rebuild` is not specified
- **filterfields** (*str, optional*) – The filter options specified via `buildtest buildspec find --filter` that contains list of key value pairs for filtering buildspecs
- **formatfields** (*str, optional*) – The format options used for formatting table. The format option is a comma separated list of format fields specified via `buildtest buildspec find --format`
- **roots** (*list, optional*) – List of directories to search for buildspecs. This argument contains value of `buildtest buildspec find --roots`
- **headers** (*bool, optional*) – Option to control whether header are printed in terse output. This argument contains value of `buildtest buildspec find --no-header`
- **terse** (*bool, optional*) – Enable terse mode when printing output. In this mode we don't print output in table format instead output is printed in parseable format. This option can be specified via `buildtest buildspec find --terse`

table

`filter_fields = ['type', 'executor', 'tags', 'buildspec']`

`default_format_fields = ['name', 'type', 'executor', 'tags', 'description']`

`format_fields`

`get_cache(self)`

Returns cache file as loaded dictionary

`load_paths(self)`

Add all paths to search for buildspecs. We must read configuration file and check property `buildspec_roots` for list of directories to search. We check all directories exist, if any fail we don't add them to path. If `load_default_buildspecs: True` is set we will add the default buildspecs which are `tutorials` and `general_tests` directory.

build(self)

This method will build builds spec cache file. If user requests to rebuild cache we remove the file and recreate cache. If cache file exists, we simply load from cache

_discover_buildspecs(self)

This method retrieves buildspecs based on `self.paths` which is a list of directory paths to search. If `--root` is specified we process each argument and recursively find all .yaml files

_write_buildspec_cache(self)

This method is responsible for writing buildspec cache to file

_validate_buildspecs(self, buildspecs)

Given a list of buildspec files, validate each buildspec using `buildtest.buildsystem.parser.BuildspecParser` class and return a list of valid buildspecs. Any invalid buildspecs are added to separate list

Parameters **buildspecs** – A list of buildspec to validate

get_names(self)

Return a list of test names found in buildspec cache. We only return test names for valid buildspecs

lookup_buildspec_by_name(self, name)

Given an input test name, return corresponding buildspec file found in the cache.

Parameters **name** (*str*) – Name of test to query in cache

Returns Return path to buildspec that contains name of test

build_cache(self)

This method will rebuild the buildspec cache file by recursively searching all .yaml files specified by input argument paths which is a list of directory roots. The buildspecs are validated and cache file is updated

_check_filter_fields(self)

This method checks filter fields are valid. The filter fields are specified as `buildtest buildspec find --filter <KEY1>=<VAL1>,<KEY2>=<VAL2>,...`

Raises `BuildTestError` – If there is an invalid filter field

_check_format_fields(self)

This method will check if all format fields are valid. Format fields are passed as comma separated fields: `--format field1,field2,field3,...`

Raises `BuildTestError` – If there is an invalid format field

_filter_buildspecs(self, executor, tags, schema_type)

This method will return a boolean True/False that determines if buildspec test entry is skipped as part of filter process. The filter are done based on executor, tags, type field. True indicates test needs to be skipped.

Parameters

- **executor** (*str*) – executor property from buildspec
- **tags** (*str*) – tags property from buildspec
- **schema_type** (*str*) – type property from buildspec

Returns Return True if there is **no** match otherwise returns False

Return type `bool`

find_buildspecs(self)

This method will find buildspecs based on cache content. We skip any tests based on executor filter, tag filter or type filter and build a table of tests that will be printed using `print_buildspecs` method.

Raises **BuildTestError** – Raises exception if input buildspec for buildtest buildspec find --filter buildspec is invalid path or directory or buildspec not found in cache.

get_valid_buildspecs(self)

Return a list of valid buildspecs

get_invalid_buildspecs(self)

Return a list of invalid buildspecs

get_unique_tags(self)

Return a list of unique tags.

get_unique_executors(self)

Return a list of unique executors.

get_maintainers(self)

Return a list of maintainers.

get_paths(self)

Return a list of search paths

tag_breakdown(self)

This method will return a breakdown of tags by test names.

executor_breakdown(self)

This method will return a dictionary with breakdown of executors by test names.

test_breakdown_by_buildspec(self)

This method will return a dictionary with breakdown of buildspecs by test names.

print_buildspecfiles(self, terse=None, header=None)

This method implements buildtest buildspec find --buildspec which reports all buildspec files in cache.

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if --terse option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If --terse option is not specified this argument has no effect. This argument holds the value of --no-header option

print_tags(self, terse=None, header=None)

This method implements buildtest buildspec find --tags which reports a list of unique tags from all buildspecs in cache file.

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if --terse option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If --terse option is not specified this argument has no effect. This argument holds the value of --no-header option

print_executors(self, terse=None, header=None)

This method implements buildtest buildspec find --executors which reports all executors from cache.

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if --terse option is specified otherwise will print output in table format

- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If `--terse` option is not specified this argument has no effect. This argument holds the value of `--no-header` option

print_by_executors(*self*, *terse=None*, *header=None*)

This method prints executors by tests and implements buildtest builds spec find `--group-by-executor` command

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if `--terse` option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If `--terse` option is not specified this argument has no effect. This argument holds the value of `--no-header` option

print_by_tags(*self*, *terse=None*, *header=None*)

This method prints tags by tests and implements buildtest builds spec find `--group-by-tags` command

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if `--terse` option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If `--terse` option is not specified this argument has no effect. This argument holds the value of `--no-header` option

print_buildspecs(*self*, *terse=None*, *header=None*)

Print builds spec table. This method is typically called when running buildtest builds spec find or options with `--filter` and `--format`.

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if `--terse` option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If `--terse` option is not specified this argument has no effect. This argument holds the value of `--no-header` option

print_maintainer(*self*, *terse=None*, *header=None*)

This method prints maintainers from builds spec cache file which implements buildtest builds spec find `--maintainers` command.

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if `--terse` option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If `--terse` option is not specified this argument has no effect. This argument holds the value of `--no-header` option

print_maintainers_by_buildspecs(*self*, *terse=None*, *header=None*)

This method prints maintainers breakdown by buildspecs. This method implements buildtest builds spec find `--maintainers-by-buildspecs`

Parameters

- **terse** (*bool*, *optional*) – This argument will print output in terse format if `--terse` option is specified otherwise will print output in table format
- **header** (*bool*, *optional*) – This argument controls whether header will be printed in terse format. If `--terse` option is not specified this argument has no effect. This argument holds the value of `--no-header` option

print_invalid_buildspecs(*self*, *error=None*)

Print invalid buildspecs from cache file. This method implements command `buildtest buildspec find invalids`

Parameters **error** (*bool*, *optional*) – Display error messages for invalid buildspecs. Default is False where we only print list of invalid buildspecs

static print_filter_fields()

This method prints filter fields available for buildspec cache. This method implements command `buildtest buildspec find --helpfilter`

static print_format_fields()

This method prints format fields available for buildspec cache. This method implements command `buildtest buildspec find --helpformat`

print_paths(*self*)

This method print buildspec paths, this implements command `buildtest buildspec find --paths`

`buildtest.cli.buildspec.show_buildspecs`(*name*, *configuration*)

This is the entry point for `buildtest buildspec show` command which will print content of buildspec based on name of test.

Parameters

- **name** (*str*) – Name of test
- **configuration** (`buildtest.config.SiteConfiguration`) – Instance of SiteConfiguration class

`buildtest.cli.buildspec.buildspec_validate`(*configuration*, *buildspecs=None*,
excluded_buildspecs=None, *tags=None*, *executors=None*)

Entry point for `buildtest buildspec validate`. This method is responsible for discovering buildspec with same options used for building buildspecs that includes `--buildspec`, `--exclude`, `--tag`, and `--executor`. Upon discovery we pass each buildspec to `BuildspecParser` class to validate buildspec and report any errors during validation which is raised as exceptions.

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of SiteConfiguration class which is the loaded buildtest configuration used for validating the buildspecs.
- **buildspecs** (*list*, *optional*) – List of paths to buildspec file which can be a file or directory. This option is specified via `buildtest buildspec validate --buildspec`
- **excluded_buildspecs** (*list*, *optional*) – List of excluded buildspecs which can be a file or directory. This option is specified via `buildtest buildspec validate --exclude`
- **tags** (*list*, *optional*) – List of tag names to search for buildspec to validate. This option is specified via `buildtest buildspec validate --tag`
- **executors** (*list*, *optional*) – List of executor names to search for buildspecs to validate. This option is specified via `buildtest buildspec validate --executor`

`buildtest.cli.buildspec.summarize_buildspec_cache`(*configuration*)

Prints summary of buildspec cache which is run via command `buildtest buildspec summary`

Parameters `configuration` (`buildtest.config.SiteConfiguration`) – instance of type Site-Configuration

`buildtest.cli.buildspec.buildspec_find(args, configuration)`

Entry point for buildtest buildspec find command

Parameters

- **args** (`dict`) – Parsed arguments from `ArgumentParser.parse_args`
- **configuration** (`buildtest.config.SiteConfiguration`) – instance of type SiteConfiguration

`buildtest.cli.cd`

Module Contents

Functions

<code>change_directory(test)</code>	Given a test name we will change directory to root of test for last test run. This
-------------------------------------	------------------------------------------------------------------------------------

`buildtest.cli.cd.change_directory(test)`

Given a test name we will change directory to root of test for last test run. This method implements command `buildtest cd`

Parameters `test` (`str`) – Name of test found in test report. The test is specified via `buildtest cd <test>`

`buildtest.cli.cdash`

Module Contents

Functions

<code>cdash_cmd(args, default_configuration=None, open_browser=True)</code>	This method is entry point for buildtest cdash command which implements uploading
<code>upload_test_cdash(build_name, configuration, site=None, report_file=None)</code>	This method is responsible for reading report file and pushing results to CDASH

`buildtest.cli.cdash.cdash_cmd(args, default_configuration=None, open_browser=True)`

This method is entry point for buildtest cdash command which implements uploading results to CDASH server and command line interface to open CDASH project.

Parameters

- **args** (`dict`) – Parsed arguments from `ArgumentParser.parse_args`
- **default_configuration** (`buildtest.config.SiteConfiguration`, *optional*) – The loaded default configuration which is an instance of SiteConfiguration class
- **open_browser** (`bool`, *optional*) – boolean to control if we open page in web browser using `webbrowser.open()`. This is enabled by default, but can be turned off especially when

running regression test where we don't want to see the page

`buildtest.cli.cdash.upload_test_cdash(build_name, configuration, site=None, report_file=None)`

This method is responsible for reading report file and pushing results to CDASH server. User can specify cdash settings in configuration file or pass them in command line. The command `buildtest cdash upload` will upload results to CDASH.

Shown below is an example output. In this example **demo** is the build name that shows up in CDASH result.

```
bash-3.2$ buildtest cdash upload demo
Reading configuration file: /Users/siddiq90/Documents/GitHubDesktop/buildtest/
↳ buildtest/settings/config.yml
Reading report file: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/report.
↳ json
build name: demo
site: generic
stamp: 20210908-1445-Experimental
MD5SUM: 078202fdea13860d50eff19a9ea737db
PUT STATUS: 200
You can view the results at: https://my.cdash.org/viewTest.php?buildid=2063736
```

Parameters

- **build_name** (*str*) – build name that shows up in CDASH
- **configuration** (`buildtest.config.SiteConfiguration`) – Instance of SiteConfiguration class that contains the configuration file
- **site** (*str*) – Site name that shows up in CDASH
- **report** (*str*) – Path to report file when uploading results. This is specified via `buildtest cdash upload -r` command

`buildtest.cli.clean`

Module Contents

Functions

<code>clean(configuration, yes)</code>	Entry point for <code>buildtest clean</code> command which will clean up directories and configuration files generated by <code>buildtest</code> .
----------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------

`buildtest.cli.clean.clean(configuration, yes)`

Entry point for `buildtest clean` command which will clean up directories and configuration files generated by `buildtest`. User will be prompted for series of question with (Y/N) to select response to each action which can be ignored by passing `--yes` option.

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of SiteConfiguration class
- **yes** (*bool*) – boolean to control whether user response is required before cleaning up tasks.

buildtest.cli.compilers

Module Contents

Classes

BuildtestCompilers

param settings_file Specify an alternate settings file to use when finding compilers

Functions

compiler_cmd(args, configuration)

compiler_find(args, configuration) This method implements buildtest config compilers find which detects

buildtest.cli.compilers.**compiler_cmd**(args, configuration)

buildtest.cli.compilers.**compiler_find**(args, configuration)

This method implements buildtest config compilers find which detects new compilers based on module names defined in configuration. If system has Lmod we use Lmodule API to detect the compilers. For environment-modules we search for all modules in current \$MODULEPATH.

class buildtest.cli.compilers.**BuildtestCompilers**(configuration, settings_file=None, debug=False)

Parameters

- **settings_file** – Specify an alternate settings file to use when finding compilers
- **settings_file** – str, optional
- **compilers** (*dict*) – compiler section from buildtest configuration.

compiler_table

find_compilers(self)

This method returns compiler modules discovered depending on your module system. If you have Lmod system we use spider utility to detect modules, this is leveraging Lmodule API. If you have environment-modules we parse output of `module av -t`.

Returns return a list of compiler modules detected based on module key name.

Return type *dict*

_validate_modules(self, module_dict)

This method will validate modules by running module load test for all discovered modules specified in parameter discovered_modules. This method returns a list of modules that were valid, if all tests pass we return the same list. A module test pass if we get a returncode 0.

_update_compiler_section(self)

This method will update the compiler section by adding new compilers if found

Returns Updated compiler section for buildtest configuration

Return type `dict`

print_json(self)

Prints compiler section in JSON, this implements `buildtest config compilers --json`

print_yaml(self)

Prints compiler section in YAML, this implements `buildtest config compilers --yaml`

list(self)

Return all compilers defined in buildtest configuration

print_compilers(self)

This method implements `buildtest config compilers` which prints all compilers from buildtest configuration

buildtest.cli.config

Module Contents

Functions

<code>config_cmd(args, configuration)</code>	Entry point for <code>buildtest config</code> command. This method will invoke other methods depending on input argument.
<code>view_system(configuration)</code>	This method implements command <code>buildtest config systems</code> which displays
<code>validate_config(configuration)</code>	This method implements <code>buildtest config validate</code> which attempts to
<code>view_configuration(configuration)</code>	Display content of buildtest configuration file. This implements command <code>buildtest config view</code>
<code>view_executors(configuration, buildexecutor, json_format=False, yaml_format=False, disabled=False, invalid=False)</code>	Display executors from buildtest configuration. This implements <code>buildtest config executors</code> command.
<code>view_summary(configuration, buildtestsystem=None)</code>	This method implements <code>buildtest config summary</code> option. In this method

`buildtest.cli.config.config_cmd(args, configuration)`

Entry point for `buildtest config` command. This method will invoke other methods depending on input argument.

Parameters

- **args** (`dict`) – Parsed arguments from `ArgumentParser.parse_args`
- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of `SiteConfiguration` class

`buildtest.cli.config.view_system(configuration)`

This method implements command `buildtest config systems` which displays system details from configuration file in table format.

Parameters **configuration** (`buildtest.config.SiteConfiguration`) – An instance of `SiteConfiguration` class

`buildtest.cli.config.validate_config(configuration)`

This method implements `buildtest config validate` which attempts to validate buildtest schema file `settings.schema.json`. If it's not validate an exception is raised which could be `jsonschema.exceptions.ValidationError` or `buildtest.exceptions.ConfigurationError`.

If configuration is valid buildtest print something as follows.

```
bash-3.2$ buildtest config validate
/Users/siddiq90/Documents/GitHubDesktop/buildtest/buildtest/settings/config.yml is_
↪ valid
```

If there is an error validating configuration file, buildtest will print error message reported by exception

Parameters `configuration` (`buildtest.config.SiteConfiguration`) – An instance of Site-Configuration class

Raises `SystemExit` – If exception is raised during validating configuration file.

`buildtest.cli.config.view_configuration(configuration)`

Display content of buildtest configuration file. This implements command `buildtest config view`

`buildtest.cli.config.view_executors(configuration, buildexecutor, json_format=False, yml_format=False, disabled=False, invalid=False)`

Display executors from buildtest configuration. This implements `buildtest config executors` command.

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of SiteConfiguration class
- **buildexecutor** (`buildtest.executors.setup.BuildExecutor`) – An instance of BuildExecutor class
- **json_format** (`bool`) – Display output in json format which is specified via `buildtest config executors --json`
- **yml_format** (`bool`) – Display output in yaml format which is specified via `buildtest config executors --yaml`
- **disabled** (`bool`) – Display list of disabled executors which is specified via `buildtest config executors --disabled`
- **invalid** (`bool`) – Display list of invalid executors which is specified via `buildtest config executors --invalid`

`buildtest.cli.config.view_summary(configuration, buildtestsystem=None)`

This method implements `buildtest config summary` option. In this method we will display a summary of System Details, Buildtest settings, Schemas, Repository details, Buildsspecs files and test names.

Parameters

- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of SiteConfiguration class
- **buildexecutor** (`buildtest.executors.setup.BuildExecutor`) – An instance of BuildExecutor class

`buildtest.cli.edit`

Module Contents

Functions

<code>edit_buildspec(buildspec, configuration)</code>	Open buildspec in editor and validate buildspec with parser. This method is invoked by command <code>buildtest edit</code> .
-------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------

`buildtest.cli.edit.edit_buildspec(buildspec, configuration)`

Open buildspec in editor and validate buildspec with parser. This method is invoked by command `buildtest edit`.

Parameters

- **buildspec** (*str*) – Path to buildspec file to edit
- **configuration** (`buildtest.config.SiteConfiguration`) – An instance of SiteConfiguration class

`buildtest.cli.help`

Module Contents

Functions

<code>print_build_help()</code>	This method will print help message for command <code>buildtest help build</code>
<code>print_buildspec_help()</code>	This method will print help message for command <code>buildtest help buildspec</code>
<code>print_config_help()</code>	This method will print help message for command <code>buildtest help config</code>
<code>print_inspect_help()</code>	This method will print help message for command <code>buildtest help inspect</code>
<code>print_report_help()</code>	This method will print help message for command <code>buildtest help report</code>
<code>print_edit_help()</code>	This method will print help message for command <code>buildtest help edit</code>
<code>print_history_help()</code>	This method will print help message for command <code>buildtest help history</code>
<code>print_cdash_help()</code>	This method will print help message for command <code>buildtest help cdash</code>
<code>print_schema_help()</code>	This method will print help message for command <code>buildtest help schema</code>
<code>print_path_help()</code>	This method will print help message for command <code>buildtest help schema</code>
<code>buildtest_help(command)</code>	Entry point for <code>buildtest help</code> which display a summary of how to use buildtest commands

`buildtest.cli.help.print_build_help()`

This method will print help message for command `buildtest help build`

`buildtest.cli.help.print_buildspec_help()`

This method will print help message for command `buildtest help buildspec`

`buildtest.cli.help.print_config_help()`

This method will print help message for command `buildtest help config`

`buildtest.cli.help.print_inspect_help()`

This method will print help message for command `buildtest help inspect`

`buildtest.cli.help.print_report_help()`

This method will print help message for command `buildtest help report`

`buildtest.cli.help.print_edit_help()`

This method will print help message for command `buildtest help edit`

`buildtest.cli.help.print_history_help()`

This method will print help message for command `buildtest help history`

`buildtest.cli.help.print_cdash_help()`

This method will print help message for command `buildtest help cdash`

`buildtest.cli.help.print_schema_help()`

This method will print help message for command `buildtest help schema`

`buildtest.cli.help.print_path_help()`

This method will print help message for command `buildtest help schema`

`buildtest.cli.help.buildtest_help(command)`

Entry point for `buildtest help` which display a summary of how to use `buildtest` commands

Parameters `command` (*str*) – Name of `buildtest` command specified by `buildtest help <command>`

`buildtest.cli.history`

Module Contents

Functions

<code>build_history(args)</code>	This is the entry point for command <code>buildtest build history</code> command which reports
<code>sorted_alphanumeric(data)</code>	This method is used for alpha numeric sorting of files.
<code>list_builds(header=None, terse=None)</code>	This method is entry point for <code>buildtest history list</code> which prints all previous builds
<code>query_builds(build_id, log_option)</code>	This method is called when user runs <i>buildtest history query</i> which will

Attributes

logger

`buildtest.cli.history.logger`

`buildtest.cli.history.build_history(args)`

This is the entry point for command `buildtest build history` command which reports

Parameters `args` (*dict*) – Parsed arguments from `ArgumentParser.parse_args`

`buildtest.cli.history.sorted_alphanumeric(data)`

This method is used for alpha numeric sorting of files.

Parameters `data` – A list of history files to sort alpha numerically

Returns sorted list of history files alphanumerically

Return type `list`

`buildtest.cli.history.list_builds(header=None, terse=None)`

This method is entry point for `buildtest history list` which prints all previous builds stored in **BUILD_HISTORY_DIR**. Each directory has a `build.json` file that stores content of each build that was run by `buildtest build`.

Parameters

- **header** (*bool*, *optional*) – Control whether header columns are displayed with terse format
- **terse** (*bool*, *optional*) – Print output in terse format

`buildtest.cli.history.query_builds(build_id, log_option)`

This method is called when user runs `buildtest history query` which will report the `build.json` and logfile.

Parameters

- **build_id** (*int*) – Build Identifier which is used for querying history file. The identifier is an integer starting from 0
- **log_option** (*bool*) – Option to control whether log file is opened in editor. This is specified via `buildtest history query -l <id>`

`buildtest.cli.inspect`

This module implements methods for `buildtest inspect` command that can be used to retrieve test record from report file in JSON format.

Module Contents

Functions

<code>inspect_cmd(args)</code>	Entry point for buildtest inspect command
<code>inspect_list(report, terse=None, header=None, builder=None)</code>	This method list an output of test id, name, and buildspec file from the report cache. The default
<code>inspect_query(report, args)</code>	Entry point for buildtest inspect query command.
<code>inspect_buildspec(report, input_buildspecs, all_records)</code>	This method implements command buildtest inspect buildspec
<code>inspect_by_name(report, names, all_records)</code>	Implements command buildtest inspect name which will print all test records by given name in JSON format.
<code>inspect_by_id(report, args)</code>	This method implements buildtest inspect id command

`buildtest.cli.inspect.inspect_cmd(args)`
Entry point for buildtest inspect command

Parameters `args (dict)` – Parsed arguments from `ArgumentParser.parse_args`

`buildtest.cli.inspect.inspect_list(report, terse=None, header=None, builder=None)`
This method list an output of test id, name, and buildspec file from the report cache. The default behavior is to display output in table format though this can be changed with terse format which will display in parseable format. This method implements command buildtest inspect list

Parameters

- **report** (`str`) – Path to report file
- **terse** (`bool`, *optional*) – Print output in terse format
- **header** (`bool`, *optional*) – Determine whether to print header in terse format.
- **builder** (`bool`, *optional*) – Print output in builder format which can be done via buildtest inspect list --builder

`buildtest.cli.inspect.inspect_query(report, args)`
Entry point for buildtest inspect query command.

Parameters

- **args** (`dict`) – Parsed arguments from `ArgumentParser.parse_args`
- **report** (`str`) – Path to report file

`buildtest.cli.inspect.inspect_buildspec(report, input_buildspecs, all_records)`
This method implements command buildtest inspect buildspec

Parameters

- **report** (`str`) – Path to report file
- **input_buildspecs** (`list`) – List of buildspecs to search in report file. This is specified as positional arguments to buildtest inspect buildspec
- **all_records** (`bool`) – Determine whether to display all records for every test that matches the buildspec. By default we retrieve the latest record.

`buildtest.cli.inspect.inspect_by_name(report, names, all_records)`

Implements command `buildtest inspect name` which will print all test records by given name in JSON format.

Parameters

- **report** (*str*) – Path to report file
- **names** (*list*) – List of test names to search in report file. This is specified as positional arguments to `buildtest inspect name`
- **all_records** (*bool*) – Determine whether to display all records for every test that matches the buildspec. By default we retrieve the latest record.

`buildtest.cli.inspect.inspect_by_id(report, args)`

This method implements `buildtest inspect id` command

`buildtest.cli.path`

Module Contents

Functions

<code>path_cmd(name, testpath=None, outfile=None, errfile=None, buildscript=None, stagedir=None)</code>	This is the entry point for <code>buildtest path</code> command which will display path
---------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------

`buildtest.cli.path.path_cmd(name, testpath=None, outfile=None, errfile=None, buildscript=None, stagedir=None)`

This is the entry point for `buildtest path` command which will display path variables for a given test name. If no options are specified we retrieve the root directory where test is installed for the latest run for test. One can specify a specific test ID by specifying backslash / folowed by test identifier.

Shown below are some examples

```
# get test root for latest run of 'circle_area'
bash-3.2$ buildtest path circle_area
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/
↳python-shell/circle_area/ac3d8bd8

# get test root for identifier that starts with 'e37'
bash-3.2$ buildtest path circle_area/e37
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/
↳python-shell/circle_area/e371dcb8

# get output file for test circle_area
bash-3.2$ buildtest path -o circle_area
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/
↳python-shell/circle_area/ac3d8bd8/circle_area.out

# get error file for test circle_area
bash-3.2$ buildtest path -e circle_area
/Users/siddiq90/Documents/GitHubDesktop/buildtest/var/tests/generic.local.python/
↳python-shell/circle_area/ac3d8bd8/circle_area.err
```


Parameters

- **name** (*str*) – Name of test to search in report file
- **testpath** (*bool*) – Retrieve path to testpath for a given test
- **outfile** (*bool*) – Retrieve path output file for a given test
- **errfile** (*bool*) – Retrieve path to error file for a given test
- **buildscript** (*bool*) – Retrieve path to build script for a given test
- **stagedir** (*bool*) – Retrieve path to stage directory for a given test

buildtest.cli.report

Module Contents

Classes

Report

param report_file Full path to report file to read

Functions

<i>is_int</i> (val)	Check if input is an integer by running <code>int()</code> . If its successful we
<i>report_cmd</i> (args)	Entry point for <code>buildtest report</code> command
<i>report_summary</i> (report)	This method will print summary for report file which can be retrieved via <code>buildtest report summary</code> command

Attributes

logger

buildtest.cli.report.logger

buildtest.cli.report.is_int(val)

Check if input is an integer by running `int()`. If its successful we return **True** otherwise returns **False**

class buildtest.cli.report.**Report**(*report_file=None, filter_args=None, format_args=None, latest=None, oldest=None*)

Parameters

- **report_file** (*str, optional*) – Full path to report file to read

- **filter_args** (*str*, *optional*) – A comma separated list of Key=Value pair for filter arguments via `buildtest report --filter`
- **format** (*str*, *optional*) – A comma separated list of format fields for altering report table. This is specified via `buildtest report --format`
- **latest** (*bool*, *optional*) – Fetch latest run for all tests discovered. This is specified via `buildtest report --latest`
- **oldest** (*bool*, *optional*) – Fetch oldest run for all tests discovered. This is specified via `buildtest report --oldest`

`format_fields = ['buildspec', 'command', 'compiler', 'endtime', 'errfile', 'executor', 'full_id', 'hostname', ...]`

`filter_fields = ['buildspec', 'name', 'executor', 'state', 'tags', 'returncode']`

`display_table`

`reportfile(self)`

Return full path to report file

`get(self)`

Return raw content of report file

`_check_filter_fields(self)`

This method will validate filter fields `buildtest report --filter` by checking if field is valid filter field. If one specifies an invalid filter field, we will raise an exception

Raises *BuildTestError* – Raise exception if its in invalid filter field. If returncode is not an integer we raise exception

`_check_format_fields(self)`

Check all format arguments (`--format`) are valid, the arguments are specified in format (`--format key1=val1,key2=val2`). We make sure each key is valid format field.

Raises *BuildTestError* – If format field is not valid

`load(self)`

This method is responsible for loading report file. If file not found or report is empty dictionary we raise an error. The report file is loaded if its valid JSON file and returns as dictionary containing entire report of all tests.

Raises *SystemExit* – If report file doesn't exist or path is not a file. If the report file is empty upon loading we raise an error.

`filter_buildspecs_from_report(self)`

This method filters the report table input filter `--filter buildspec`. If entry found in buildspec cache we add to list

`_filter_by_names(self, name)`

Filter test by name of test. This method will return True if record should be processed, otherwise returns False

Parameters *name* (*str*) – Name of test to filter

`_filter_by_tags(self, test)`

This method will return a boolean (True/False) to check if test should be skipped from report. Given an input test, we check if test has 'tags' property in buildspec and if tagnames specified by `--filter tags` are found in the test. If there is a match we return False. A True indicates the test will be filtered out.

Parameters *test* (*dict*) – Test recorded loaded as dictionary

_filter_by_executor(*self*, *test*)

Filters test by `executor` property given input parameter `buildtest report --filter executor:<executor>`. If there is **no** match we return `True` otherwise returns `False`.

Parameters `test` (*dict*) – Test recorded loaded as dictionary

_filter_by_state(*self*, *test*)

This method filters test by `state` property based on input parameter `buildtest report --filter state:<STATE>`. If there is **no** match we return `True` otherwise returns `False`.

Parameters `test` (*dict*) – Test recorded loaded as dictionary

_filter_by_returncode(*self*, *test*)

Returns `True/False` if test is filtered by returncode. We will get input returncode in filter field via `buildtest report --filter returncode:<CODE>` with one in test and if there is a match we return `True` otherwise returns `False`.

Parameters `test` (*dict*) – Test recorded loaded as dictionary

process_report(*self*)

print_format_fields(*self*)

Displays list of format field which implements command `buildtest report --helpformat`

print_filter_fields(*self*)

Displays list of help filters which implements command `buildtest report --helpfilter`

print_report(*self*, *terse=None*, *noheader=None*)

This method will print report table after processing report file. By default we print output in table format but this can be changed to terse format which will print output in parseable format.

Parameters

- **terse** (*bool*, *optional*) – Print output in terse format
- **noheader** (*bool*, *optional*) – Determine whether to print header in terse format

In this example, we display output in tabular format which works with `--filter` and `--format` option.

```
bash-3.2$ buildtest report --filter name=root_disk_usage --format name,state,
↳returncode
Reading report file: /Users/siddiq90/Documents/GitHubDesktop/buildtest/var/
↳report.json
```

name	state	returncode
root_disk_usage	PASS	0
root_disk_usage	PASS	0
root_disk_usage	PASS	0

In terse format each output is separated by PIPE symbol (`|`). The first row contains headers followed by content of the report.

```
bash-3.2$ buildtest report --filter name=root_disk_usage --format name,state,
↳returncode --terse
name|state|returncode
```

(continues on next page)

(continued from previous page)

```
root_disk_usage|PASS|0
root_disk_usage|PASS|0
root_disk_usage|PASS|0
```

You can avoid printing the header table by specifying *--no-header* option

```
bash-3.2$ buildtest report --filter name=root_disk_usage --format name,state,
↪returncode --terse --no-header
root_disk_usage|PASS|0
root_disk_usage|PASS|0
root_disk_usage|PASS|0
```

latest_testid_by_name(*self*, *name*)

Given a test name return test id of latest run

Parameters *name* (*str*) – Name of test to search in report file and retrieve corresponding test id

get_names(*self*)

Return a list of test names from report file

get_buildspecs(*self*)

Return a list of buildspecs in report file

get_testids(*self*)

Return a list of test ids from the report file

_testid_lookup(*self*)

Return a dict where *key* represents full id of test and value is a dictionary containing two values *name* and *buildspec* property which contains name of test and path to buildspec file.

builder_names(*self*)

Return a list of builder names in builder format which is in the form: *<NAME>/<TESTID>*.

breakdown_by_test_names(*self*)

Returns a dictionary with number of test runs, pass test and fail test by testname

fetch_records_by_ids(*self*, *testids*)

Fetch a test record given a list of test identifier.

Parameters *testids* (*list*) – A list of test IDs to search in report file and retrieve JSON record for each test.

buildtest.cli.report.report_cmd(*args*)

Entry point for buildtest report command

buildtest.cli.report.report_summary(*report*)

This method will print summary for report file which can be retrieved via `buildtest report summary` command

buildtest.cli.schema**Module Contents****Functions**

<i>schema_cmd</i> (args)	This method implements command <code>buildtest schema</code> which shows a list
--------------------------	---------------------------------------------------------------------------------

buildtest.cli.schema.schema_cmd(args)

This method implements command `buildtest schema` which shows a list of schemas, their json content and list of schema examples. The input `args` is an instance of `argparse` class that contains user selection via command line.

Parameters `args` (*dict*) – Parsed arguments from `ArgumentParser.parse_args`

Package Contents**Functions**

<i>single_kv_string</i> (val)	This method is used for filter field in <code>buildtest build --filter</code> .
<i>handle_kv_string</i> (val)	This method is used as type field in <code>-filter</code> argument in <code>buildtest buildspect find</code> .
<i>positive_number</i> (value)	Checks if input value is positive value and within range of 1-50. This method
<i>get_parser</i> ()	
<i>edit_menu</i> (subparsers)	
<i>history_menu</i> (subparsers)	This method builds the command line menu for <code>buildtest history</code> command
<i>build_menu</i> (subparsers)	This method implements command line menu for <code>buildtest build</code> command.
<i>buildspec_menu</i> (subparsers)	This method implements <code>buildtest buildspect</code> command
<i>config_menu</i> (subparsers)	This method adds <code>argparse</code> argument for <code>buildtest config</code>
<i>report_menu</i> (subparsers)	This method implements the <code>buildtest report</code> command options
<i>inspect_menu</i> (subparsers)	This method builds argument for <code>buildtest inspect</code> command
<i>schema_menu</i> (subparsers)	This method builds menu for <code>buildtest schema</code>
<i>cdash_menu</i> (subparsers)	This method builds arguments for <code>buildtest cdash</code> command.

Attributes

BUILDTEST_COPYRIGHT

BUILDTEST_VERSION

BUILD_REPORT

schema_table

`buildtest.cli.BUILDTEST_COPYRIGHT` = Copyright (c) 2021, The Regents of the University of California, through Lawrence Berkeley...

`buildtest.cli.BUILDTEST_VERSION` = 0.11.0

`buildtest.cli.BUILD_REPORT`

`buildtest.cli.schema_table`

`buildtest.cli.single_kv_string(val)`

This method is used for filter field in `buildtest build --filter`. This method returns a dict of key/value pair where input must be a single key/value pair

Parameters `val (str)` – input value

Returns dictionary of key/value pairs

Return type `dict`

`buildtest.cli.handle_kv_string(val)`

This method is used as type field in `-filter` argument in `buildtest buildspect find`. This method returns a dict of key,value pair where input is in format `key1=val1,key2=val2,key3=val3`

Parameters

- `val (bool)` – input value
- **multiple_keys** – `multiple_keys` is a boolean to determine if key/value pair accepts multiple key/value arguments

Returns dictionary of key/value pairs

Return type `dict`

`buildtest.cli.positive_number(value)`

Checks if input value is positive value and within range of 1-50. This method is used for `-rebuild` option

`buildtest.cli.get_parser()`

`buildtest.cli.edit_menu(subparsers)`

`buildtest.cli.history_menu(subparsers)`

This method builds the command line menu for `buildtest history` command

`buildtest.cli.build_menu(subparsers)`

This method implements command line menu for `buildtest build` command.

`buildtest.cli.buildspec_menu(subparsers)`

This method implements `buildtest buildspect` command

`buildtest.cli.config_menu(subparsers)`
 This method adds argparse argument for `buildtest config`

`buildtest.cli.report_menu(subparsers)`
 This method implements the `buildtest report` command options

`buildtest.cli.inspect_menu(subparsers)`
 This method builds argument for `buildtest inspect` command

`buildtest.cli.schema_menu(subparsers)`
 This method builds menu for `buildtest schema`

`buildtest.cli.cdash_menu(subparsers)`
 This method builds arguments for `buildtest cdash` command.

buildtest.executors

Submodules

buildtest.executors.base

BuildExecutor: manager for test executors

Module Contents

Classes

<i>BaseExecutor</i>	The BaseExecutor is an abstract base class for all executors.
---------------------	---------------------------------------------------------------

class `buildtest.executors.base.BaseExecutor(name, settings, site_configs)`

The BaseExecutor is an abstract base class for all executors.

Initiate a base executor, meaning we provide a name (also held by the BuildExecutor base that holds it) and the loaded dictionary of config opts to parse.

Parameters

- **name** (*str*) – name of executor
- **setting** (*dict*) – setting for a given executor defined in configuration file
- **site_configs** (`buildtest.config.SiteConfiguration`) – Instance of SiteConfiguration class

type = `base`

add_builder(*self*, *builder*)

Add builder object to `self.builders` only if its of type `BuilderBase`

get_builder(*self*)

Return a list of builders

load(*self*)

Load a particular configuration based on the name. This method should set defaults for the executor, and will vary based on the class.

run(*self*)

The run step basically runs the build. This is run after setup so we are sure that the builder is defined. This is also where we set the result to return.

__str__(*self*)

Return str(*self*).

__repr__(*self*)

Return repr(*self*).

buildtest.executors.cobalt

This method implements CobaltExecutor class which defines how cobalt executor submit job to Cobalt scheduler.

Module Contents

Classes

<i>CobaltExecutor</i>	The CobaltExecutor class is responsible for submitting jobs to Cobalt Scheduler.
<i>CobaltJob</i>	The CobaltJob class performs operation on cobalt job upon job submission such

Attributes

<i>logger</i>

buildtest.executors.cobalt.logger

class buildtest.executors.cobalt.**CobaltExecutor**(*name, settings, site_configs, max_pend_time=None*)
Bases: *buildtest.executors.base.BaseExecutor*

The CobaltExecutor class is responsible for submitting jobs to Cobalt Scheduler. The class implements the following methods:

- **load**: load Cobalt executors from configuration file
- **dispatch**: submit Cobalt job to scheduler
- **poll**: poll Cobalt job via qstat and retrieve job state
- **gather**: gather job record including output, error, exit code

Initiate a base executor, meaning we provide a name (also held by the BuildExecutor base that holds it) and the loaded dictionary of config opts to parse.

Parameters

- **name** (*str*) – name of executor
- **setting** (*dict*) – setting for a given executor defined in configuration file
- **site_configs** (*buildtest.config.SiteConfiguration*) – Instance of SiteConfiguration class

type = cobalt

load(*self*)

Load the a Cobalt executor configuration from buildtest settings.

launcher_command(*self*)

dispatch(*self*, *builder*)

This method is responsible for dispatching job to Cobalt Scheduler by invoking `builder.run()` which runs the build script. If job is submitted to scheduler, we get the JobID and pass this to CobaltJob class. At job submission, cobalt will report the output and error file which can be retrieved using `qstat`. We retrieve the cobalt job record using `builder.job.gather()`.

Parameters *builder* (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

poll(*self*, *builder*)

This method is responsible for polling Cobalt job by invoking the builder method `builder.job.poll()`. We check the job state and existence of output file. If file exists or job is complete, we gather the results and return from function. If job is pending we check if job time exceeds `max_pend_time` time limit and cancel job.

Parameters *builder* (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

gather(*self*, *builder*)

This method is responsible for moving output and error file in the run directory. We need to read `<JOBID>.cobaltlog` file which contains output of exit code by performing a regular expression (`exit code of.)(\d+)(\;)`). The cobalt log file will contain a line: **task completed normally with an exit code of 0; initiating job cleanup and removal**

Parameters *builder* (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

class buildtest.executors.cobalt.CobaltJob(*jobID*)

Bases: `buildtest.executors.job.Job`

The CobaltJob class performs operation on cobalt job upon job submission such as polling job, gather job record, cancel job. We also retrieve job state and determine if job is pending, running, complete, suspended.

is_pending(*self*)

Return True if job is pending otherwise returns False. When cobalt receives job it is in `starting` followed by `queued` state. We check if job is in either state.

is_running(*self*)

Return True if job is running otherwise returns False. Cobalt job state for running job is marked as `running`

is_complete(*self*)

Return True if job is complete otherwise returns False. Cobalt job state for completed job is marked as `exiting`

is_suspended(*self*)

Return True if job is suspended otherwise returns False. Cobalt job state for suspended is marked as `user_hold`

is_cancelled(*self*)

Return True if job is cancelled otherwise returns False. Job state is `cancelled` which is set by class `cancel` method

cobalt_log(*self*)

Return job `cobalt.log` file

output_file(self)

Return job output file

error_file(self)

Return job error file

exitcode(self)

Return job exit code

poll(self)

Poll job by running `qstat -l --header State <jobid>` which retrieves job state.

gather(self)

Gather Job state by running `qstat -lf <jobid>` which retrieves all fields. The output is in text format which is parsed into key/value pair and stored in a dictionary. This method will return a dict containing the job record

```
$ qstat -lf 347106
JobID: 347106
  JobName      : hold_job
  User         : shahzebsiddiqui
  WallTime     : 00:10:00
  QueuedTime   : 00:13:14
  RunTime      : N/A
  TimeRemaining : N/A
```

cancel(self)

Cancel job by running `qdel <jobid>`. This method is called if job timer exceeds `max_pend_time` if job is pending.

buildtest.executors.job

Module Contents

Classes

<i>Job</i>	This is a base class for holding job level data and common methods for used
------------	-----------------------------------------------------------------------------

class buildtest.executors.job.Job(*jobID*)

This is a base class for holding job level data and common methods for used for batch job submission.

state(self)

Return job state

get(self)

Return Job ID

abstract is_pending(self)

Check if job is in pending state

abstract is_running(self)

Check if job is in running state

abstract is_suspended(self)

Check if job is in suspended state

```
abstract cancel(self)
    Cancel job

abstract poll(self)
    Poll job and update job state.
```

buildtest.executors.local

This module implements the LocalExecutor class responsible for submitting jobs to localhost. This class is called in class BuildExecutor when initializing the executors.

Module Contents

Classes

<i>LocalExecutor</i>	The LocalExecutor class is responsible for running tests locally for
----------------------	----------------------------------------------------------------------

class buildtest.executors.local.**LocalExecutor**(name, settings, site_configs)

Bases: *buildtest.executors.base.BaseExecutor*

The LocalExecutor class is responsible for running tests locally for bash, sh, csh and python shell. The LocalExecutor runs the tests and gathers the output and error results and writes to file.

Initiate a base executor, meaning we provide a name (also held by the BuildExecutor base that holds it) and the loaded dictionary of config opts to parse.

Parameters

- **name** (*str*) – name of executor
- **setting** (*dict*) – setting for a given executor defined in configuration file
- **site_configs** (*buildtest.config.SiteConfiguration*) – Instance of SiteConfiguration class

type = local

load(self)

Load a particular configuration based on the name. This method should set defaults for the executor, and will vary based on the class.

check(self)

Check if shell binary is available.

Raises *SystemExit* – If path to shell is invalid

run(self, builder)

This method is responsible for running test for LocalExecutor which runs test locally. We keep track of metadata in builder.metadata that keeps track of run result. The output and error file are written to filesystem.

Parameters **builder** (*buildtest.buildsystem.base.BuilderBase*) – An instance object of BuilderBase type

buildtest.executors.lsf

This module implements the LSFExecutor class responsible for submitting jobs to LSF Scheduler. This class is called in class BuildExecutor when initializing the executors.

Module Contents

Classes

<i>LSFExecutor</i>	The LSFExecutor class is responsible for submitting jobs to LSF Scheduler.
<i>LSFJob</i>	This is a base class for holding job level data and common methods for used

Attributes

<i>logger</i>

buildtest.executors.lsf.logger

class buildtest.executors.lsf.LSFExecutor(*name, settings, site_configs, max_pend_time=None*)

Bases: *buildtest.executors.base.BaseExecutor*

The LSFExecutor class is responsible for submitting jobs to LSF Scheduler. The LSFExecutor performs the following steps

- **load**: load lsf configuration from buildtest configuration file
- **dispatch**: dispatch job to scheduler and acquire job ID
- **poll**: wait for LSF jobs to finish
- **gather**: Once job is complete, gather job data

Initiate a base executor, meaning we provide a name (also held by the BuildExecutor base that holds it) and the loaded dictionary of config opts to parse.

Parameters

- **name** (*str*) – name of executor
- **setting** (*dict*) – setting for a given executor defined in configuration file
- **site_configs** (*buildtest.config.SiteConfiguration*) – Instance of SiteConfiguration class

type = lsf

load(*self*)

Load the a LSF executor configuration from buildtest settings.

launcher_command(*self*)

This command returns the launcher command and any options specified in configuration file. This is useful when generating the build script in the BuilderBase class

dispatch(*self*, *builder*)

This method is responsible for dispatching job to scheduler and extracting job ID by applying a `re.search` against output at onset of job submission. If job id is not retrieved due to job failure or unable to match regular expression we mark job incomplete by invoking `buildtest.buildsystem.base.BuilderBase.incomplete`()` method and return from method.

If we have a valid job ID we invoke `buildtest.executors.lsf.LSFJob` class given the job id to poll job and store this into `builder.job` attribute.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

poll(*self*, *builder*)

Given a builder object we poll the job by invoking builder method `builder.job.poll()` return state of job. If job is suspended or pending we stop timer and check if timer exceeds `max_pend_time` value which could be defined in configuration file or passed via command line `--max-pend-time`

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

gather(*self*, *builder*)

Gather Job detail after completion of job by invoking the builder method `builder.job.gather()`. We retrieve exit code, output file, error file and update builder metadata.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

class `buildtest.executors.lsf.LSFJob(jobID)`

Bases: `buildtest.executors.job.Job`

This is a base class for holding job level data and common methods for used for batch job submission.

is_pending(*self*)

Check if Job is pending which is reported by LSF as PEND. Return True if there is a match otherwise returns False

is_running(*self*)

Check if Job is running which is reported by LSF as RUN. Return True if there is a match otherwise returns False

is_complete(*self*)

Check if Job is complete which is in DONE state. Return True if there is a match otherwise return False

is_suspended(*self*)

Check if Job is in suspended state which could be in any of the following states: [PSUSP, USUSP, SSUSP]. We return True if job is in one of the states otherwise return False

is_failed(*self*)

Check if Job failed. We return True if job is in EXIT state otherwise return False

output_file(*self*)

Return job output file

error_file(*self*)

Return job error file

exitcode(*self*)

Return job exit code

poll(*self*)

Given a job id we poll the LSF Job by retrieving its job state, output file, error file and exit code. We run the following commands to retrieve following states

- Job State: `bjobs -noheader -o 'stat' <JOBID>`
- Exit Code: `bjobs -noheader -o 'EXIT_CODE' <JOBID>`

gather(*self*)

This method will retrieve the output and error file for a given jobID using the following commands.

```
$ bjobs -noheader -o 'output_file' 70910
hold_job.out
```

```
$ bjobs -noheader -o 'error_file' 70910
hold_job.err
```

We will gather job record at onset of job completion by running `bjobs -o '<format1> <format2>' <jobid> -json`. The format fields extracted from job are the following:

- “job_name”
- “stat”
- “user”
- “user_group”
- “queue”
- “proj_name”
- “pids”
- “exit_code”
- “from_host”
- “exec_host”
- “submit_time”
- “start_time”
- “finish_time”
- “nthreads”
- “exec_home”
- “exec_cwd”
- “output_file”
- “error_file”

Shown below is the output format and we retrieve the job records defined in **RECORDS** property

```
$ bjobs -o 'job_name stat user user_group queue proj_name pids exit_code from_
↪host exec_host submit_time start_time finish_time nthreads exec_home exec_cwd_
↪output_file error_file' 58652 -json
{
  "COMMAND": "bjobs",
  "JOBS": 1,
  "RECORDS": [
    {
      "JOB_NAME": "hold_job",
      "STAT": "PSUSP",
```

(continues on next page)

(continued from previous page)

```

        "USER": "shahzebsiddiqui",
        "USER_GROUP": "GEN014ECPCI",
        "QUEUE": "batch",
        "PROJ_NAME": "GEN014ECPCI",
        "PIDS": "",
        "EXIT_CODE": "",
        "FROM_HOST": "login1",
        "EXEC_HOST": "",
        "SUBMIT_TIME": "May 28 12:45",
        "START_TIME": "",
        "FINISH_TIME": "",
        "NTHREADS": "",
        "EXEC_HOME": "",
        "EXEC_CWD": "",
        "OUTPUT_FILE": "hold_job.out",
        "ERROR_FILE": "hold_job.err"
    }
]
}

```

cancel(self)

Cancel LSF Job by running `bkill <jobid>`. This is called if job has exceeded *max_pend_time* limit during poll stage.

buildtest.executors.pbs

This module implements PBSExecutor class that defines how executors submit job to PBS Scheduler

Module Contents**Classes**

<i>PBSExecutor</i>	The PBSExecutor class is responsible for submitting jobs to PBS Scheduler.
<i>PBSJob</i>	The PBSJob models a PBS Job with helper methods to retrieve job state, check if job is running/pending/suspended. We have methods

Attributes

<i>logger</i>

buildtest.executors.pbs.logger

class buildtest.executors.pbs.**PBSExecutor**(name, settings, site_configs, max_pend_time=None)

Bases: *buildtest.executors.base.BaseExecutor*

The PBSExecutor class is responsible for submitting jobs to PBS Scheduler. The class implements the following methods:

- **load**: load PBS executors from configuration file
- **dispatch**: submit PBS job to scheduler
- **poll**: poll PBS job via qstat and retrieve job state
- **gather**: gather job result
- **cancel**: cancel job if it exceeds max pending time

Initiate a base executor, meaning we provide a name (also held by the BuildExecutor base that holds it) and the loaded dictionary of config opts to parse.

Parameters

- **name** (*str*) – name of executor
- **setting** (*dict*) – setting for a given executor defined in configuration file
- **site_configs** (*buildtest.config.SiteConfiguration*) – Instance of SiteConfiguration class

type = pbs

poll_cmd = qstat

load(*self*)

Load the a PBS executor configuration from buildtest settings.

launcher_command(*self*)

dispatch(*self*, *builder*)

This method is responsible for dispatching PBS job, get JobID and start record metadata in builder object. If job failed to submit we check returncode and exit with failure. After we submit job, we start timer and record when job was submitted and poll job once to get job details and store them in builder object.

Parameters **builder** (*buildtest.buildsystem.base.BuilderBase*) – An instance object of BuilderBase type

poll(*self*, *builder*)

This method is responsible for polling PBS job which will update the job state. If job is complete we will gather job result. If job is pending we will stop timer and check if pend time exceeds max pend time for executor. If so we will cancel the job.

Parameters **builder** (*buildtest.buildsystem.base.BuilderBase*) – An instance object of BuilderBase type

gather(*self*, *builder*)

This method is responsible for gather job results including output and error file and complete metadata for job which is stored in the builder object. We will retrieve job exitcode which corresponds to test returncode.

Parameters **builder** (*buildtest.buildsystem.base.BuilderBase*) – An instance object of BuilderBase type

class buildtest.executors.pbs.PBSJob(*jobID*)

Bases: *buildtest.executors.job.Job*

The PBSJob models a PBS Job with helper methods to retrieve job state, check if job is running/pending/suspended. We have methods to poll job state, gather job results upon completion and cancel job.

See <https://www.altair.com/pdfs/pbsworks/PBSReferenceGuide2021.1.pdf> section 8.1 for list of Job State Codes

is_pending(self)

Return True if job is pending. A pending job is in state Q.

is_running(self)

Return True if job is running. A completed job is in state R.

is_complete(self)

Return True if job is complete. A completed job is in state F.

is_suspended(self)

Return True if job is suspended which would be in one of these states H, U, S.

output_file(self)

Return output file of job

error_file(self)

Return error file of job

exitcode(self)

Return exit code of job

success(self)

This method determines if job was completed successfully and returns True if exit code is 0.

According to <https://www.altair.com/pdfs/pbsworks/PBSAdminGuide2021.1.pdf> section 14.9 Job Exit Status Codes we have the following

- Exit Code: $X < 0$ - Job could not be executed
- Exit Code: $0 \leq X < 128$ - Exit value of Shell or top-level process
- Exit Code: $X \geq 128$ - Job was killed by signal
- Exit Code: $X == 0$ - Job executed was a successful

fail(self)

Return True if there is a job failure which would be if exit code is not 0

poll(self)

This method will poll the PBS Job by running `qstat -x -f -F json <jobid>` which will report job data in JSON format that can be parsed to extract the job state. In PBS the active job state can be retrieved by reading property `job_state` property. Shown below is an example output

```
[pbsuser@pbs tests]$ qstat -x -f -F json 157.pbs
{
  "timestamp":1630683518,
  "pbs_version":"19.0.0",
  "pbs_server":"pbs",
  "Jobs":{
    "157.pbs":{
      "Job_Name":"pbs_hold_job",
      "Job_Owner":"pbsuser@pbs",
      "job_state":"H",
      "queue":"workq",
      "server":"pbs",
      "Checkpoint":"u",
      "ctime":"Fri Aug 20 23:14:08 2021",
      "Error_Path":"pbs:/tmp/GitHubDesktop/buildtest/var/tests/generic.
↪pbs.workq/hold/pbs_hold_job/da6d5b57/stage/pbs_hold_job.e157", (continues on next page)
```

(continued from previous page)

```

        "Hold_Types": "u",
        "Join_Path": "n",
        "Keep_Files": "n",
        "Mail_Points": "a",
        "mtime": "Fri Aug 20 23:14:08 2021",
        "Output_Path": "pbs:/tmp/GitHubDesktop/buildtest/var/tests/generic.
↪pbs.workq/hold/pbs_hold_job/da6d5b57/stage/pbs_hold_job.o157",
        "Priority": 0,
        "qtime": "Fri Aug 20 23:14:08 2021",
        "Rerunable": "True",
        "Resource_List": {
            "ncpus": 1,
            "nodect": 1,
            "nodes": 1,
            "place": "scatter",
            "select": "1:ncpus=1",
            "walltime": "00:02:00"
        },
        "substate": 20,
        "Variable_List": {
            "PBS_O_HOME": "/home/pbsuser",
            "PBS_O_LOGNAME": "pbsuser",
            "PBS_O_PATH": "/tmp/GitHubDesktop/buildtest/bin:/tmp/github/
↪buildtest/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/opt/pbs/
↪bin:/home/pbsuser/.local/bin:/home/pbsuser/bin",
            "PBS_O_MAIL": "/var/spool/mail/pbsuser",
            "PBS_O_SHELL": "/bin/bash",
            "PBS_O_WORKDIR": "/tmp/GitHubDesktop/buildtest/var/tests/generic.
↪pbs.workq/hold/pbs_hold_job/da6d5b57/stage",
            "PBS_O_SYSTEM": "Linux",
            "PBS_O_QUEUE": "workq",
            "PBS_O_HOST": "pbs"
        },
        "Submit_arguments": "-q workq /tmp/GitHubDesktop/buildtest/var/tests/
↪generic.pbs.workq/hold/pbs_hold_job/da6d5b57/stage/pbs_hold_job.sh",
        "project": "_pbs_project_default"
    }
}

```

gather(self)

This method is called once job is complete. We will gather record of job by running `qstat -x -f -F json <jobid>` and return the json object as a dict. This method is responsible for getting output file, error file and exit status of job.

cancel(self)

Cancel PBS job by running `qdel <jobid>`.

`buildtest.executors.poll`

Module Contents

Classes

PollQueue

class `buildtest.executors.poll.PollQueue`(*builders, interval, buildexecutor*)

cancelled(*self*)

Return a list of cancelled builders

completed(*self*)

sleep(*self*)

poll(*self*)

Poll all pending jobs until all jobs are complete. At each poll interval, we poll each builder job state. If job is complete or failed we remove job from pending queue. In each interval we sleep and poll jobs until there is no pending jobs.

print_pending_jobs(*self*)

Print pending jobs in table format during each poll step

print_polled_jobs(*self*)

`buildtest.executors.setup`

This module is responsible for setup of executors defined in buildtest configuration. The BuildExecutor class initializes the executors and chooses the executor class (LocalExecutor, LSFExecutor, SlurmExecutor, CobaltExecutor) to call depending on executor name.

Module Contents

Classes

BuildExecutor

A BuildExecutor is responsible for initialing executors from buildtest configuration

Attributes

logger

`buildtest.executors.setup.logger`

class `buildtest.executors.setup.BuildExecutor`(*site_config*, *max_pend_time=None*)

A BuildExecutor is responsible for initialing executors from buildtest configuration file which provides a list of executors. This class keeps track of all executors and provides the following methods:

- **setup**: This method will write executor's `before_script.sh` that is sourced in each test upon calling executor.
- **run**: Responsible for invoking executor's `run` method based on builder object which is of type `BuilderBase`.
- **poll**: This is responsible for invoking `poll` method for corresponding executor from the builder object by checking job state

Initialize executors, meaning that we provide the buildtest configuration that are validated, and can instantiate each executor to be available.

Parameters

- **site_config** (`buildtest.config.SiteConfiguration`) – instance of SiteConfiguration class that has the buildtest configuration
- **max_pend_time** (*int*, *optional*) – maximum pend time in second until job is cancelled.

`__str__(self)`

Return `str(self)`.

`__repr__(self)`

Return `repr(self)`.

`list_executors(self)`

`is_local(self, executor_type)`

`is_slurm(self, executor_type)`

`is_lsf(self, executor_type)`

`is_pbs(self, executor_type)`

`is_cobalt(self, executor_type)`

`get(self, name)`

Given the name of an executor return the executor object which is of subclass of *BaseExecutor*

`_choose_executor(self, builder)`

Choose executor is called at the onset of a run and poll stage. Given a builder object we retrieve the executor property `builder.executor` of the builder and check if there is an executor object and of type *BaseExecutor*.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of *BuilderBase* type

`setup(self)`

This method creates directory `var/executors/<executor-name>` for every executor defined in buildtest configuration and write scripts `before_script.sh` if the field `before_script` is specified in executor section. This method is called after executors are initialized in the class `__init__` method.

load_builders(*self*, *builders*)

Adds builder objects into `self.builders` class variable. This method will only add objects that are instance of `BuilderBase` class.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of `BuilderBase` type

run(*self*)

This method is responsible for running the build script for each builder async and gather the results. We setup a pool of worker settings by invoking `multiprocessing.pool.Pool` and use `multiprocessing.pool.Pool.apply_sync()` method for running test async which returns an object of type `multiprocessing.pool.AsyncResult` which holds the result. Next we wait for results to arrive using `multiprocessing.pool.AsyncResult.get()` method in a infinite loop until all test results are retrieved. The return type is the same builder object which is added to list of valid builders that is returned at end of method.

get_builders(*self*)

Return a list of valid builders that were run

buildtest.executors.slurm

This module implements the `SlurmExecutor` class responsible for submitting jobs to Slurm Scheduler. This class is called in class `BuildExecutor` when initializing the executors.

Module Contents**Classes**

<i>SlurmExecutor</i>	The <code>SlurmExecutor</code> class is responsible for submitting jobs to Slurm Scheduler.
<i>SlurmJob</i>	The <code>SlurmJob</code> class models a Slurm Job ID with helper methods to perform operation against an active slurm job. The <code>SlurmJob</code> class

Attributes

<i>logger</i>

buildtest.executors.slurm.logger

class `buildtest.executors.slurm.SlurmExecutor`(*name*, *settings*, *site_configs*, *max_pend_time=None*)

Bases: `buildtest.executors.base.BaseExecutor`

The `SlurmExecutor` class is responsible for submitting jobs to Slurm Scheduler. The `SlurmExecutor` performs the following steps:

- **load**: load slurm configuration from buildtest configuration file
- **dispatch**: dispatch job to scheduler and acquire job ID
- **poll**: wait for Slurm jobs to finish, if job is pending and exceeds `max_pend_time` then cancel job
- **gather**: Once job is complete, gather job data

Initiate a base executor, meaning we provide a name (also held by the BuildExecutor base that holds it) and the loaded dictionary of config opts to parse.

Parameters

- **name** (*str*) – name of executor
- **setting** (*dict*) – setting for a given executor defined in configuration file
- **site_configs** (`buildtest.config.SiteConfiguration`) – Instance of SiteConfiguration class

type = `slurm`

load(*self*)

Load the a slurm executor configuration from buildtest settings.

launcher_command(*self*)

Return sbatch launcher command with options used to submit job

dispatch(*self*, *builder*)

This method is responsible for dispatching job to slurm scheduler and extracting job id. If job id is valid we pass the job to `buildtest.executors.slurm.SlurmJob` class and store object in `builder.job`.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

poll(*self*, *builder*)

This method is called during poll stage where we invoke `builder.job.poll()` to get updated job state. If job is pending or suspended we stop timer and check if job needs to be cancelled if time exceeds `max_pend_time` value.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

gather(*self*, *builder*)

Gather Slurm job data after job completion. In this step we call `builder.job.gather()`, and update builder metadata such as returncode, output and error file.

Parameters **builder** (`buildtest.buildsystem.base.BuilderBase`) – An instance object of BuilderBase type

class `buildtest.executors.slurm.SlurmJob(jobID, cluster=None)`

Bases: `buildtest.executors.job.Job`

The SlurmJob class models a Slurm Job ID with helper methods to perform operation against an active slurm job. The SlurmJob class can poll job to get updated job state, gather job data upon completion of test and cancel job if necessary. We can also retrieve job state and determine if job is running, pending, suspended, or cancelled. Jobs are polled via `sacct` command which can retrieve pending, running and complete jobs.

is_pending(*self*)

If job is pending return True otherwise return False. Slurm Job state for pending is PENDING.

is_running(*self*)

If job is running return True otherwise return False. Slurm will report RUNNING for job state.

is_suspended(*self*)

If job is suspended return True otherwise return False. Slurm will report SUSPENDED for job state.

is_cancelled(*self*)

If job is cancelled return True otherwise return False. Slurm will report CANCELLED for job state.

is_complete(*self*)

If job is complete return True otherwise return False. Slurm will report COMPLETED for job state.

is_failed(*self*)

If job failed return True otherwise return False. Slurm will report FAILED for job state.

is_out_of_memory(*self*)

If job is out of memory return True otherwise return False. Slurm will report OUT_OF_MEMORY for job state.

is_timeout(*self*)

If job timed out return True otherwise return False. Slurm will report TIMEOUT for job state.

complete(*self*)

This method is used for gathering job result we assume job is complete if it's in any of the following state: COMPLETED, FAILED, OUT_OF_MEMORY, TIMEOUT

state(*self*)

Return job state

workdir(*self*)

Return job work directory

exitcode(*self*)

Return job exit code

cancel(*self*)

Cancel job by running `sacancel <jobid>`. If job is specified to a slurm cluster we cancel job using `sacancel <jobid> --clusters=<cluster>`. This method is called if job exceeds *max_pend_time*.

poll(*self*)

This method will poll job via `sacct` command to get updated job state by running the following command:
`sacct -j <jobid> -o State -n -X -P`

Slurm will report the job state that can be parsed. Shown below is an example job that is PENDING state

```
$ sacct -j 46641229 -o State -n -X -P
PENDING
```

gather(*self*)

Gather job record which is called after job completion. We use `sacct` to gather job record and return the job record as a dictionary. The command we run is `sacct -j <jobid> -X -n -P -o <field1>, <field2>, ..., <fieldN>`. We retrieve the following format fields from job record:

- "Account"
- "AllocNodes"
- "AllocTRES"
- "ConsumedEnergyRaw"
- "CPUTimeRaw"
- "Elapsed"
- "End"
- "ExitCode"
- "JobID"
- "JobName"
- "NCPUS"
- "NNodes"

- “QOS”
- “ReqMem”
- “ReqNodes”
- “Start”
- “State”
- “Submit”
- “UID”
- “User”
- “WorkDir”

The output of `sacct` is parseable using the pipe symbol (`|`) and stored into a dict

```
$ sacct -j 42909266 -X -n -P -o Account,AllocNodes,AllocTRES,ConsumedEnergyRaw,  
↳ CPUTimeRaw,Elapsed,End,ExitCode,JobID,JobName,NCPUS,NNodes,QOS,ReqMem,  
↳ ReqNodes,Start,State,Submit,UID,User,WorkDir --clusters=cori  
nstaff|1|billing=272,cpu=272,energy=262,mem=87G,node=1|262|2176|00:00:08|2021-  
↳ 05-27T18:47:49|0:0|42909266|slurm_metadata|272|1|debug_knl|87Gn|1|2021-05-  
↳ 27T18:47:41|COMPLETED|2021-05-27T18:44:07|92503|siddiq90|/global/u1/s/  
↳ siddiq90/.buildtest/tests/cori.slurm.knl_debug/metadata/slurm_metadata/0/stage
```

We retrieve `ExitCode` and `WorkDir` via `sacct` command to get returncode. Slurm will write output and error file in `WorkDir` location. We run the following command below and parse the output. The `ExitCode` is in form `<exitcode>:<signal>` which is colon separated list. For more details on Slurm Exit code see https://slurm.schedmd.com/job_exit_code.html

```
$ sacct -j 46294283 --clusters=cori -X -n -P -o ExitCode,Workdir  
0:0|/global/u1/s/siddiq90/github/buildtest/var/tests/cori.slurm.knl_debug/  
↳ hostname/hostname_knl/cd39a853/stage
```

buildtest.schemas

Submodules

buildtest.schemas.defaults

Module Contents

Functions

`custom_validator`(recipe, schema)

This is a custom validator for validating JSON documents. We implement a

Attributes

here

schema_table

schema_store

resolver

`buildtest.schemas.defaults.here`

`buildtest.schemas.defaults.schema_table`

`buildtest.schemas.defaults.schema_store`

`buildtest.schemas.defaults.resolver`

`buildtest.schemas.defaults.custom_validator(recipe, schema)`

This is a custom validator for validating JSON documents. We implement a custom resolver using [RefResolver](#) to find schemas locally in order to validate buildsspecs with schema files on local filesystem. This ensures changes to schema can be done in sync with change to code base.

This method uses [Draft7Validator](#) for validating schemas. If there is an error during validation jsonschema will raise an exception of type `jsonschema.exceptions.ValidationError`

Parameters

- **recipe** (*dict*) – Loaded test recipe as YAML document
- **schema** (*dict*) – Schema document loaded in JSON format

Raises `jsonschema.exceptions.ValidationError` – if recipe fails to validate with schema

`buildtest.schemas.utils`

Utility and helper functions for schemas.

Module Contents

Functions

<code>load_schema(path)</code>	Load a json schema file, the file extension must be .schema.json
<code>load_recipe(path)</code>	Load a yaml recipe file. The file must be in .yaml extension for buildtest to load.

Attributes

here

`buildtest.schemas.utils.here`

`buildtest.schemas.utils.load_schema(path)`

Load a json schema file, the file extension must be `.schema.json`

Parameters `path` (*str*) – Path to schema file

Returns Return loaded schema as JSON document

Return type *dict*

Raises `SystemExit` – If filepath doesn't exist or schema file doesn't ends in `.schema.json`

`buildtest.schemas.utils.load_recipe(path)`

Load a yaml recipe file. The file must be in `.yaml` extension for buildtest to load.

Parameters `path` (*str*) – The full path to buildspect recipe

Returns a dict containing buildspect that is defined in YAML format

Return type *dict*

Raises `SystemExit` – If filepath doesn't exist or doesn't end in `.yaml` extension

`buildtest.utils`

Submodules

`buildtest.utils.command`

Module Contents

Classes

<i>Capturing</i>	capture output from stdout and stderr into capture object.
<i>BuildTestCommand</i>	Class method to invoke shell commands and retrieve output and error.

class `buildtest.utils.command.Capturing`

capture output from stdout and stderr into capture object. This is based off of github.com/vsoch/gridtest but modified to write files. The stderr and stdout are set to temporary files at the init of the capture, and then they are closed when we exit. This means expected usage looks like:

```
with Capturing() as capture:
    process = subprocess.Popen(...)
```

And then the output and error are retrieved from reading the files: and exposed as properties to the client: `capture.out`, `capture.err`

And cleanup means deleting these files, if they exist.

```

__enter__(self)
set_stdout(self)
set_stderr(self)
__exit__(self, *args)
property out(self)
    Return content of output stream if file exists otherwise returns empty string
property err(self)
    Return content of error stream if file exists otherwise returns empty string.
cleanup(self)
    This method will remove stdout and stderr file upon reading both streams

```

```
class buildtest.utils.command.BuildTestCommand(cmd=None)
```

Class method to invoke shell commands and retrieve output and error. This class is inspired and derived from utils functions in <https://github.com/vsoch/scif>

The initializer method will initialize class variables and check input argument *cmd* and make sure command is in a list format.

Parameters *cmd* (*str*, *optional*) – Input shell command

```
set_command(self, cmd)
```

parse is called when a new command is provided to ensure we have a list. We don't check that the executable is on the path, as the initialization might not occur in the runtime environment.

```
execute(self)
```

Execute a system command and return output and error.

```
returncode(self)
```

Returns the return code from shell command

Returns returncode of shell command

Return type *int*

```
decode(self, line)
```

Given a line of output (error or regular) decode using the system default, if appropriate

```
get_output(self)
```

Returns the output content from shell command

```
get_error(self)
```

Returns the error content from shell command

```
get_command(self)
```

Returns the executed command

buildtest.utils.file

This module provides some generic file and directory level operation that include the following: 1. Check if path is a File or Directory via `is_file()`, `is_dir()` 2. Create a directory via `create_dir()` 3. Walk a directory tree based on single extension using `walk_tree()` 4. Resolve path including shell and user expansion along with getting realpath to file using `resolve_path()` 5. Read and write a file via `read_file()`, `write_file()`

Module Contents

Functions

<code>is_file(fname)</code>	Check if file exist and returns True/False
<code>is_dir(dirname)</code>	Check if input directory exist and is a directory. If so return True otherwise returns False .
<code>walk_tree(root_dir, ext=None)</code>	This method will traverse a directory tree and return list of files
<code>create_dir(dirname)</code>	Create a directory if it doesn't exist. If directory contains variable
<code>resolve_path(path, exist=True)</code>	This method will resolve a file path to account for shell expansion and resolve paths in
<code>read_file(filepath)</code>	This method is used to read a file and return content of file.
<code>write_file(filepath, content)</code>	This method is used to write an input content to a file specified by
<code>remove_file(fpath)</code>	This method is responsible for removing a file. The input path is an absolute path
<code>load_json(fname)</code>	Given a filename, resolves full path to file and loads json file. This method will

`buildtest.utils.file.is_file(fname)`

Check if file exist and returns True/False

Parameters `fname` (*str*) – file path to check

Returns True if path is a file and is a realpath otherwise returns False

Return type *bool*

`buildtest.utils.file.is_dir(dirname)`

Check if input directory exist and is a directory. If so return **True** otherwise returns **False**. We resolve path by invoking `resolve_path()`

Parameters `dirname` (*str*) – directory path to check

Returns True if directory exists otherwise returns False.

Return type *bool*

`buildtest.utils.file.walk_tree(root_dir, ext=None)`

This method will traverse a directory tree and return list of files based on extension type. This method invokes `is_dir()` to check if directory exists before traversal.

Parameters

- **root_dir** (*str*) – directory path to traverse
- **ext** (*str*) – File extension to search in traversal

Returns A list of file paths for a directory traversal based on extension type. If **ext** is **None** we retrieve all files

Return type *list*

`buildtest.utils.file.create_dir(dirname)`

Create a directory if it doesn't exist. If directory contains variable expansion (**\$HOME**) or user expansion (~), we resolve this before creating directory. If there is an error creating directory we raise an exception `BuildTestError`

Parameters `dirname` (*str*, *required*) – directory path to create

Returns creates the directory or print an exception message upon failure

Return type Catches exception of type `OSError` and raise exception `BuildTestError` or returns `None`

`buildtest.utils.file.resolve_path(path, exist=True)`

This method will resolve a file path to account for shell expansion and resolve paths in when a symlink is provided in the file. This method assumes file already exists.

Parameters

- `path` (*str*) – file path to resolve
- `exist` (*bool*) – a boolean to determine if filepath should be returned if filepath doesn't exist on filesystem.

Returns Full path to file if file exists or `exist=True` is set. We could return `None` if path is not defined or file path doesn't exist and `exist=False`

Return type *str*

Raises `BuildTestError` – If input path is not of type *str*

```
>>> a = resolve_path("$HOME/.bashrc")
>>> assert a
>>> b = resolve_path("$HOME/.bashrc1", exist=False)
>>> assert b
>>> c = resolve_path("$HOME/.bashrc1", exist=True)
>>> assert not c
```

`buildtest.utils.file.read_file(filepath)`

This method is used to read a file and return content of file. If filepath is not a string we raise an error. We run `resolve_path()` to get realpath to file and account for shell or user expansion. The return will be a valid file or `None` so we check if input is an invalid file. Finally we read the file and return the content of the file as a string.

Parameters `filepath` (*str*) – File name to read

Raises `BuildTestError` –

- if filepath is invalid
- filepath is not an instance of type *str*.
- An exception can be raised if there is an issue reading file with an exception of `IOError`

Returns content of input file

Return type *str*

`buildtest.utils.file.write_file(filepath, content)`

This method is used to write an input `content` to a file specified by `filepath`. Both `filepath` and `content` must be a *str*. An error is raised if `filepath` is not a string or a directory. If `content` is not a *str*, we return `None` since we can't write the content to file. Finally, we write the content to file and return. A successful write will return nothing otherwise an exception will occur during the write process.

Parameters

- `filepath` (*str*) – file name to write
- `content` (*str*) – content to write to file

Raises `BuildTestError` –

- filepath is not *str*

- filepath is directory via `is_dir`
- content of file is not of type `str`
- Error writing file with an exception of type `IOError`

`buildtest.utils.file.remove_file(fpath)`

This method is responsible for removing a file. The input path is an absolute path to file. We check for exceptions first, and return immediately before removing file.

Parameters `fpath` (`str`) – full path to file to remove

Raises `BuildTestError` –

- If `fpath` is not instance of `str`
- If `fpath` is not a file using `is_file()`
- An exception of type `OSError` when removing file via `os.remove()`

`buildtest.utils.file.load_json(fname)`

Given a filename, resolves full path to file and loads json file. This method will catch exception `json.JSONDecodeError` and raise an exception with useful message. If there is no error we return content of json file

Parameters `fname` (`str`) – Name of file to read and load json content

Raises `BuildTestError` – Raise exception if file is not resolved via `resolve_path()` or failure to load JSON document

`buildtest.utils.shell`

Module Contents

Classes

<code>Shell</code>	The Shell initializer takes an input shell and shell options and split
--------------------	------------------------------------------------------------------------

class `buildtest.utils.shell.Shell(shell='bash')`

The Shell initializer takes an input shell and shell options and split string by shell name and options.

Parameters `shell` (`str`) – Specify shell program and any options passed to shell. Defaults to `bash`

`valid_shells = ['bash', 'sh', 'zsh', 'csh', 'tcsh', '/bin/bash', '/bin/csh', '/bin/sh', '/bin/tcsh', ...]`

property `opts(self)`

retrieve the shell opts that are set on init, and updated with setter

property `path(self)`

This method returns the full path to shell program using `shutil.which()` If shell program is not found we raise an exception. The shebang is updated assuming path is valid which is just adding character `#!` in front of path. The return is full path to shell program. This method automatically updates the shell path when there is a change in attribute `self.name`

```
>>> shell = Shell("bash")
>>> shell.path
```

(continues on next page)

(continued from previous page)

```

'/usr/bin/bash'
>>> shell.name="sh"
>>> shell.path
'/usr/bin/sh'

```

__str__(self)

Return str(self).

__repr__(self)

Return repr(self).

get(self)

Return shell attributes as a dictionary

buildtest.utils.timer**Module Contents****Classes***Timer***exception buildtest.utils.timer.TimerError**Bases: [Exception](#)

A custom exception used to report errors in use of Timer class

Initialize self. See help(type(self)) for accurate signature.

class buildtest.utils.timer.Timer**start(self)**

Start a new timer

stop(self)

Stop the timer, and report the elapsed time

duration(self)**buildtest.utils.tools****Module Contents****Classes***Hasher*

dict() -> new empty dictionary

Functions

deep_get(dictionary, *keys)

`buildtest.utils.tools.deep_get(dictionary, *keys)`

class `buildtest.utils.tools.Hasher`

Bases: `dict`

`dict()` -> new empty dictionary `dict(mapping)` -> new dictionary initialized from a mapping object's (key, value) pairs

dict(iterable) -> new dictionary initialized as if via: `d = {}` for `k, v` in iterable:

`d[k] = v`

dict(kwargs)** -> new dictionary initialized with the name=value pairs in the keyword argument list. For example: `dict(one=1, two=2)`

Initialize self. See `help(type(self))` for accurate signature.

__missing__(self, key)

get(self, path, sep='.', default=None)

Return the value for key if key is in the dictionary, else default.

__str__(self)

Return `str(self)`.

Submodules

`buildtest.config`

Module Contents

Classes

SiteConfiguration

This class is an interface to buildtest configuration

Attributes

logger

`buildtest.config.logger`

class `buildtest.config.SiteConfiguration(settings_file=None)`

This class is an interface to buildtest configuration

_file

Path to configuration file

Type `str`

config

Loaded configuration file

Type `dict`

target_config

Loaded configuration file for a particular system

Type `dict`

disabled_executors

A list of disabled executors when checking executors

Type `list`

invalid_executors

A list of invalid executors when checking executors

Type `list`

valid_executors

A dict containing executors that are valid for each executor type.

Type `dict`

The initializer will declare class variables in its initial state and resolve path to configuration file. Once file is resolved we will load the configuration using [load\(\)](#).

Parameters `settings_file` (`str`, *optional*) – path to buildtest configuration file

load(*self*)

Loads configuration file

property `file`(*self*)

resolve(*self*)

This method will resolve path to configuration file. The order of precedence is as follows:

1. command line argument via `buildtest --config <path>`
2. User Configuration: `$HOME/.buildtest/config.yml`
3. Default Configuration: `$BUILDTEST_ROOT/buildtest/settings/config.yml`

name(*self*)

Return name of matched system from configuration file

detect_system(*self*)

This method detects which system configuration to use by checking target hostname with list of hostname entries defined in `hostnames` property. If there is a match we set `self._name` to map to system name and load the target configuration by setting `self.target_config` to the desired system configuration.

If no system is found we raise an exception.

Raises `ConfigurationError` – If there is no matching system

validate(*self*, `validate_executors=True`)

This method validates the site configuration with schema and checks executor setting.

Parameters `validate_executors` (`bool`) – Check executor settings. This is the default behavior but can be disabled

_executor_check(*self*)

Validate executors

_validate_local_executors(*self*)

Check local executor by verifying the 'shell' types are valid

_validate_lsf_executors(*self*)

This method validates all LSF executors. We check if queue is available and in Open:Active state.

_validate_slurm_executors(*self*)

This method will validate slurm executors, we check if partition, qos, and cluster fields are valid values by retrieving details from slurm configuration. These checks are performed on fields `partition`, `qos` or `cluster` if specified in executor section.

_validate_cobalt_executors(*self*)

Validate cobalt queue property by running `qstat -Ql <queue>`. If its a non-zero exit code then queue doesn't exist otherwise it is a valid queue.

_validate_pbs_executors(*self*)

Validate pbs queue property by running by checking if queue is found and queue is 'enabled' and 'started' which are two properties found in pbs queue configuration that can be retrieved using `qstat -Q -f -F json`. The output is in the following format

```
$ qstat -Q -f -F json
{
  "timestamp":1615924938,
  "pbs_version":"19.0.0",
  "pbs_server":"pbs",
  "Queue":{
    "workq":{
      "queue_type":"Execution",
      "total_jobs":0,
      "state_count":"Transit:0 Queued:0 Held:0 Waiting:0 Running:0
↳Exiting:0 Begun:0 ",
      "resources_assigned":{
        "mem":"0kb",
        "ncpus":0,
        "nodect":0
      },
      "hasnodes":"True",
      "enabled":"True",
      "started":"True"
    }
  }
}
```

buildtest.defaults

Buildtest defaults, including environment variables and paths, are defined or derived here.

Module Contents

```
buildtest.defaults.supported_type_schemas = ['script-v1.0.schema.json',
'compiler-v1.0.schema.json']
```

```
buildtest.defaults.supported_schemas
```

```
buildtest.defaults.userhome
```

```
buildtest.defaults.BUILDTEST_ROOT
```

```
buildtest.defaults.SCHEMA_ROOT
```

```
buildtest.defaults.BUILDTEST_USER_HOME
```

```
buildtest.defaults.USER_SETTINGS_FILE
```

```
buildtest.defaults.VAR_DIR
```

```
buildtest.defaults.BUILD_HISTORY_DIR
```

```
buildtest.defaults.BUILDTEST_DEFAULT_TESTDIR
```

```
buildtest.defaults.BUILDTEST_EXECUTOR_DIR
```

```
buildtest.defaults.BUILDTEST_BUILDSPEC_DIR
```

```
buildtest.defaults.BUILDSPEC_CACHE_FILE
```

```
buildtest.defaults.BUILD_REPORT
```

```
buildtest.defaults.BUILDTEST_REPORT_SUMMARY
```

```
buildtest.defaults.BUILDSPEC_DEFAULT_PATH
```

```
buildtest.defaults.DEFAULT_SETTINGS_FILE
```

```
buildtest.defaults.DEFAULT_SETTINGS_SCHEMA
```

buildtest.exceptions

Module Contents

exception buildtest.exceptions.BuildTestError(msg, *args)

Bases: `Exception`

Class responsible for error handling in buildtest. This is a sub-class of Exception class.

This class is used for printing error message when exception is raised.

Parameters

- **msg** (*str*) – message to print
- ***args** (*list*) – extra arguments to class for printing message

__str__(self)

Return str(self).

exception buildtest.exceptions.**BuildspecError**(*buildspec, msg*)

Bases: [Exception](#)

Exception if there is an issue with parsing a Buildspec or building test

Initialize self. See help(type(self)) for accurate signature.

__str__(*self*)

Return str(self).

exception buildtest.exceptions.**ExecutorError**

Bases: [Exception](#)

This class raises an error with Executor class and its operation

Initialize self. See help(type(self)) for accurate signature.

exception buildtest.exceptions.**RuntimeFailure**

Bases: [Exception](#)

The RuntimeFailure exception is raised when there is an error running test

Initialize self. See help(type(self)) for accurate signature.

exception buildtest.exceptions.**ConfigurationError**(*config, settings_file, msg*)

Bases: [Exception](#)

ConfigurationError is raised when there is an issue with buildtest configuration file

Initialize self. See help(type(self)) for accurate signature.

__str__(*self*)

Return str(self).

buildtest.log

Methods related to buildtest logging

Module Contents

Functions

<i>init_logfile</i> (logfile=FILE_LOG, debug=None)	Initialize a log file intended for a builder. This requires
--------------------------------------------------------------------	-------------------------------------------------------------

Attributes

<i>LOG_FORMATTER</i>

<i>LOG_NAME</i>

<i>FILE_LOG</i>

buildtest.log.LOG_FORMATTER = %(asctime)s [%(filename)s:%(lineno)s - %(funcName)5s()] -
[%(levelname)s] %(message)s

```
buildtest.log.LOG_NAME = buildtest
```

```
buildtest.log.FILE_LOG
```

```
buildtest.log.init_logfile(logfile=FILE_LOG, debug=None)
```

Initialize a log file intended for a builder. This requires passing the filename intended for the log (from the builder) and returns the logger. :param logfile: logfile name :type logfile: str

buildtest.main

Entry point for buildtest

Module Contents

Functions

<i>main()</i>	Entry point to buildtest.
---------------	---------------------------

```
buildtest.main.main()
    Entry point to buildtest.
```

buildtest.system

This module detects System changes defined in class BuildTestSystem.

Module Contents

Classes

<i>BuildTestSystem</i>	BuildTestSystem is a class that detects system configuration
<i>Scheduler</i>	This is a base Scheduler class used for implementing common methods for
<i>Slurm</i>	The Slurm class implements common functions to query Slurm cluster
<i>LSF</i>	The LSF class checks for LSF binaries and returns a list of LSF queues
<i>Cobalt</i>	The Cobalt class checks for Cobalt binaries and gets a list of Cobalt queues
<i>PBS</i>	The PBS class checks for Cobalt binaries and gets a list of Cobalt queues

Attributes

system

class buildtest.system.BuildTestSystem

BuildTestSystem is a class that detects system configuration

Constructor method for BuildTestSystem(). Defines all system configuration using class variable **system** which is a dictionary.

system

get(self)

check(self)

Based on the module “distro” get system details like linux distro, processor, hostname, etc...

check_scheduler(self)

Check existence of batch scheduler and if so determine which scheduler it is. Currently we support Slurm, LSF, and Cobalt we invoke each class and see if its valid state. The checks determine if scheduler binaries exist in \$PATH.

detect_module_tool(self)

Check if module tool exists, we check for Lmod or environment-modules by checking if environment variable LMOD_VERSION, MODULE_VERSION or MODULES_CMD exist. We check this with input specification in buildtest configuration. If user specifies lmod as the module tool but detected environment-modules, buildtest should pick this up and report this as part of configuration check

class buildtest.system.Scheduler

This is a base Scheduler class used for implementing common methods for detecting Scheduler details. The subclass implement specific queries that are scheduler specific. The Slurm, LSF, PBS and Cobalt class inherit from Base Class Scheduler.

logger

check(self)

Check if binaries exist binary exist in \$PATH

class buildtest.system.Slurm

Bases: [Scheduler](#)

The Slurm class implements common functions to query Slurm cluster including partitions, qos, cluster. We check existence of slurm binaries in \$PATH and return if slurm cluster is in valid state.

binaries = ['sbatch', 'sacct', 'sacctmgr', 'sinfo', 'scancel']

_get_partitions(self)

Get list of all partitions slurm partitions using sinfo -a -h -O partitionname. The output is a list of queue names

```
$ sinfo -a -h -O partitionname
system
system_shared
debug_hsw
debug_knl
jupyter
```

_get_clusters(self)

Get list of slurm clusters by running `sacctmgr list cluster -P -n format=Cluster`. The output is a list of slurm clusters something as follows

```
$ sacctmgr list cluster -P -n format=Cluster
cori
escori
```

_get_qos(self)

Retrieve a list of all slurm qos by running `sacctmgr list qos -P -n format=Name`. The output is a list of qos. Shown below is an example output

```
$ sacctmgr list qos -P -n format=Name
normal
premium
low
serialize
scavenger
```

class buildtest.system.LSF

Bases: [Scheduler](#)

The LSF class checks for LSF binaries and returns a list of LSF queues

binaries = ['bsub', 'bqueues', 'bkill', 'bjobs']

_get_queues(self)

Return json dictionary of available LSF Queues and their queue states. The command we run is the following: `bqueues -o 'queue_name status' -json` which returns a JSON record of all queue details.

```
$ bqueues -o 'queue_name status' -json
{
  "COMMAND": "bqueues",
  "QUEUES": 2,
  "RECORDS": [
    {
      "QUEUE_NAME": "batch",
      "STATUS": "Open:Active"
    },
    {
      "QUEUE_NAME": "test",
      "STATUS": "Open:Active"
    }
  ]
}
```

class buildtest.system.Cobalt

Bases: [Scheduler](#)

The Cobalt class checks for Cobalt binaries and gets a list of Cobalt queues

binaries = ['qsub', 'qstat', 'qdel', 'nodelist', 'showres', 'partlist']

_get_queues(self)

Get all Cobalt queues by running `qstat -Ql` and parsing output

class buildtest.system.PBS

Bases: [Scheduler](#)

The PBS class checks for Cobalt binaries and gets a list of Cobalt queues

```
binaries = ['qsub', 'qstat', 'qdel', 'qstart', 'qhold', 'qmgr']
```

```
_get_queues(self)
```

Get queue configuration using qstat -Q -f -F json and retrieve a list of queues.

```
buildtest.system.system
```

Package Contents

```
buildtest.BUILDTEST_VERSION = 0.11.0
```

```
buildtest.__version__
```

```
buildtest.BUILDTEST_COPYRIGHT = Copyright (c) 2021, The Regents of the University of  
California, through Lawrence Berkeley...
```

3.14 Buildtest Command Reference

buildtest is a HPC testing framework for building and running tests.

```
usage: buildtest [options] [COMMANDS]
```

3.14.1 Named Arguments

-V, --version	show program's version number and exit
-c, --config	Specify Path to Configuration File
-d, --debug	Print debug messages to screen
	Default: False
--color	Possible choices: on, off
	Enable or disable color
	Default: "on"

3.14.2 COMMANDS

Possible choices: build, bd, buildspec, bc, config, cg, report, rt, inspect, it, history, hy, edit, et, schema, cdash, cd, clean, path, docs, schemadocs, help, h

3.14.3 Sub-commands:

build (bd)

Build and Run test

```
buildtest build [-h] [-b BUILDSPEC] [-x EXCLUDE] [-e EXECUTOR] [-t TAGS] [-f FILTER] [--
↪helpfilter]
                [--disable-executor-check] [-k] [--max-pend-time MAX_PEND_TIME] [--poll-
↪interval POLL_INTERVAL]
                [--rebuild REBUILD] [-r REPORT] [--retry RETRY] [-s {parse,build}] [--
↪testdir TESTDIR]
```

discover

select buildsspecs

-b, --buildspec	Specify a buildspec (file or directory) to build. A buildspec must end in '.yaml' extension.
-x, --exclude	Exclude one or more buildsspecs (file or directory) from processing. A buildspec must end in '.yaml' extension.
-e, --executor	Discover buildsspecs by executor name found in buildspec cache
-t, --tags	Discover buildsspecs by tags found in buildspec cache

filter

Filter tests

-f, --filter	Filter buildspec based on tags, type, or maintainers. Usage: <code>--filter key1=val1,key2=val2</code>
--helpfilter	Show available filter fields used with <code>--filter</code> option Default: False

extra

All extra options

--disable-executor-check	Disable executor check during configuration check. By default these checks are enforced for Local, Slurm, PBS, LSF, and Cobalt Executor. Default: True
-k, --keep-stage-dir	Keep stage directory after job completion. Default: False
--max-pend-time	Specify Maximum Pending Time (sec) for job before cancelling job. This only applies for batch job submission.
--poll-interval	Specify Poll Interval (sec) for polling batch jobs
--rebuild	Rebuild test X number of times. Must be a positive number between [1-50]

-r, --report	Specify a report file where tests will be written.
--retry	Retry failed jobs Default: 1
-s, --stage	Possible choices: parse, build control behavior of buildtest build
--testdir	Specify a custom test directory where to write tests. This overrides configuration file and default location.

buildspec (bc)

Buildspec Interface

```
buildtest buildspec [-h] ...
```

subcommands

Find buildspec from cache file

Possible choices: find, summary, show, validate

Sub-commands:

find

Query information from buildspecs cache

```
buildtest buildspec find [-h] [-b] [-e] [--group-by-tags] [--group-by-executor] [-m] [-
↪mb] [-p] [-t] [--filter FILTER]
                        [--format FORMAT] [--helpfilter] [--helpformat] [-n] [--terse]
↪[-r] [--root ROOT]
                        ...
```

Positional Arguments

Possible choices: invalid

Named Arguments

-r, --rebuild	Rebuild buildspec cache and find all buildspecs again Default: False
--root	Specify root buildspecs (directory) path to load buildspecs into buildspec cache.

filter and format

filter and format options

--filter	Filter buildspec cache with filter fields in format <code>--filter key1=val1,key2=val2</code>
--format	Format buildspec cache with format fields in format <code>--format field1,field2,...</code>
--helpfilter	Show Filter fields for <code>--filter</code> option for filtering buildspec cache output Default: False
--helpformat	Show Format fields for <code>--format</code> option for formatting buildspec cache output Default: False

terse

terse options

-n, --no-header	Print output without header in terse output Default: False
--terse	Print output in machine readable format Default: False

query

query options to retrieve from buildspec cache

-b, --buildspec	Get all buildspec files from cache Default: False
-e, --executors	get all unique executors from buildspecs Default: False
--group-by-tags	Group tests by tag name Default: False
--group-by-executor	Group tests by executor name Default: False
-m, --maintainers	Get all maintainers for all buildspecs Default: False
-mb, --maintainers-by-buildspecs	Show maintainers breakdown by buildspecs Default: False
-p, --paths	print all root buildspec paths Default: False
-t, --tags	List all available tags Default: False

Sub-commands:

invalid

Show invalid buildsspecs

```
buildtest buildspec find invalid [-h] [-e]
```

Named Arguments

-e, --error	Show error messages
	Default: False

summary

Print summary of buildspec cache

```
buildtest buildspec summary [-h]
```

show

Show content of buildspec file

```
buildtest buildspec show [-h] name
```

Positional Arguments

name	Show content of buildspec based on test name
-------------	----------------------------------------------

validate

Validate buildspecs with JSON Schema

```
buildtest buildspec validate [-h] [-b BUILDSPEC] [-x EXCLUDE] [-e EXECUTOR] [-t TAG]
```

Named Arguments

-b, --buildspec	Specify path to buildspec (file, or directory) to validate
-x, --exclude	Specify path to buildspec to exclude (file or directory) during validation
-e, --executor	Specify buildspecs by executor name to validate
-t, --tag	Specify buildspecs by tag name to validate

config (cg)

Query buildtest configuration

```
buildtest config [-h] ...
```

subcommands

Query information from buildtest configuration file

Possible choices: compilers, executors, summary, systems, validate, view

Sub-commands:

compilers

Search compilers

```
buildtest config compilers [-h] [-j] [-y] ...
```

Named Arguments

-j, --json	List compiler details in JSON format Default: False
-y, --yaml	List compiler details in YAML format Default: False

subcommands

Find new compilers and add them to detected compiler section

Possible choices: find

Sub-commands:

find

Find compilers

```
buildtest config compilers find [-h] [-d]
```

Named Arguments

-d, --debug	Display Debugging output when finding compilers
	Default: False

executors

Query executors from buildtest configuration

```
buildtest config executors [-h] [-j | -y | -d | -i]
```

Named Arguments

-j, --json	View executor in JSON format
	Default: False
-y, --yaml	View executors in YAML format
	Default: False
-d, --disabled	Show disabled executors
	Default: False
-i, --invalid	Show invalid executors
	Default: False

summary

Provide summary of buildtest settings.

```
buildtest config summary [-h]
```

systems

List all available systems

```
buildtest config systems [-h]
```

validate

Validate buildtest settings file with schema.

```
buildtest config validate [-h]
```

view

View Buildtest Configuration File

```
buildtest config view [-h]
```

report (rt)

Query test report

```
buildtest report [-h] [--filter FILTER] [--format FORMAT] [--helpfilter] [--helpformat]
↪ [--latest] [--oldest] [-n]
    [-r REPORT] [-t]
    ...
```

Named Arguments

--filter	Filter report by filter fields. The filter fields must be a key=value pair and multiple fields can be comma separated in the following format: <code>--filter key1=val1,key2=val2</code> . For list of filter fields run: <code>--helpfilter</code> .
--format	format field for printing purposes. For more details see <code>--helpformat</code> for list of available fields. Fields must be separated by comma (usage: <code>--format <field1>,<field2>,...</code>)
--helpfilter	List available filter fields to be used with <code>--filter</code> option Default: False
--helpformat	List of available format fields Default: False
--latest	Retrieve latest record of particular test Default: False
--oldest	Retrieve oldest record of particular test Default: False
-n, --no-header	Don't print headers column used with terse option (<code>--terse</code>). Default: False
-r, --report	Specify a report file to read Default: <code>"/home/docs/checkouts/readthedocs.org/user_builds/buildtest/checkouts/v0.11.0/var/report.json"</code>
-t, --terse	Print output in machine readable format Default: False

subcommands

Fetch test results from report file and print them in table format

Possible choices: clear, list, summary

Sub-commands:

clear

delete report file

```
buildtest report clear [-h]
```

list

List all report files

```
buildtest report list [-h]
```

summary

Summarize test report

```
buildtest report summary [-h]
```

inspect (it)

Inspect a test based on NAME or ID

```
buildtest inspect [-h] [-r REPORT] ...
```

Named Arguments

-r, --report Specify a report file to load when inspecting test

subcommands

Inspect Test result based on Test ID or Test Name

Possible choices: buildspec, id, name, query, list

Sub-commands:**buildspec**

Inspect a test based on buildspec

```
buildtest inspect buildspec [-h] [-a] [buildspec [buildspec ...]]
```

Positional Arguments

buildspec	List of buildspecs to query
------------------	-----------------------------

Named Arguments

-a, --all	Fetch all records for a given test
	Default: False

id

Specify a Test ID

```
buildtest inspect id [-h] [id [id ...]]
```

Positional Arguments

id	Test ID
-----------	---------

name

Specify name of test

```
buildtest inspect name [-h] [-a] [name [name ...]]
```

Positional Arguments

name	Name of test
-------------	--------------

Named Arguments

-a, --all Fetch all test records for a given test name
Default: False

query

Query fields from record

```
buildtest inspect query [-h] [-b] [-d {first,last,all}] [-e] [-o] [-t] [name [name ...]]
```

Positional Arguments

name Name of test

Named Arguments

-b, --buildscript Print build script
Default: False

-d, --display Possible choices: first, last, all
Determine how records are fetched, by default it will report the last record of the test.
Default: “last”

-e, --error Print error file
Default: False

-o, --output Print output file
Default: False

-t, --testpath Print content of testpath
Default: False

list

List all test names, ids, and corresponding buildsspecs

```
buildtest inspect list [-h] [-n] [-t] [-b]
```

Named Arguments

-n, --no-header	Print output without header in terse format (<code>--terse</code>) Default: False
-t, --terse	Print output in terse format Default: False
-b, --builder	List test in builder format Default: False

history (hy)

Query build history

```
buildtest history [-h] ...
```

subcommands

Query build history file

Possible choices: list, query

Sub-commands:

list

List a summary of all builds

```
buildtest history list [-h] [-n] [-t]
```

Named Arguments

-n, --no-header	Do not print header columns in terse output (<code>--terse</code>) Default: False
-t, --terse	Print output in machine readable format Default: False

query

Query information for a particular build

```
buildtest history query [-h] [-l] id
```

Positional Arguments

id Select a build ID

Named Arguments

-l, --log Display logfile for corresponding build id
Default: False

edit (et)

Edit a buildspec and validate with schema file

```
buildtest edit [-h] buildspec
```

Positional Arguments

buildspec Open buildspec in editor and validate upon closing file

schema

List schema contents and examples

```
buildtest schema [-h] [-e] [-j] [-n Schema Name]
```

Named Arguments

-e, --example Show schema examples
Default: False

-j, --json Display json schema file
Default: False

-n, --name Possible choices: global.schema.json, definitions.schema.json, settings.schema.json, compiler-v1.0.schema.json, spack-v1.0.schema.json, script-v1.0.schema.json
show schema by name (e.g., script)

cdash

Upload test to CDASH server

```
buildtest cdash [-h] ...
```

subcommands

buildtest CDASH integration

Possible choices: view, upload

Sub-commands:

view

Open CDASH project in webbrowser

```
buildtest cdash view [-h] [--url URL]
```

Named Arguments

--url Specify a url to CDASH project

upload

Upload Test to CDASH server

```
buildtest cdash upload [-h] [-r REPORT] [--site SITE] buildname
```

Positional Arguments

buildname Specify Build Name reported in CDASH

Named Arguments

-r, --report Path to report file to upload test results

--site Specify site name reported in CDASH

cd

change directory to root of test given a test name

```
buildtest cd [-h] test
```

Positional Arguments

test	Change directory to root of test for last run of test.
-------------	--------------------------------------------------------

clean

Remove all generate files from buildtest including test directory, log files, report file, buildspect cache, history files.

```
buildtest clean [-h] [-y]
```

Named Arguments

-y, --yes	Confirm yes for all prompts Default: False
------------------	-----------------------------------------------

path

Show path attributes for a given test

```
buildtest path [-h] [-t | -o | -e | -b | --stagedir] name
```

Positional Arguments

name	Name of test
-------------	--------------

Named Arguments

-t, --testpath	Show path to test script Default: False
-o, --outfile	Show path to output file Default: False
-e, --errfile	Show path to error file Default: False
-b, --buildscript	Show path to build script Default: False
--stagedir	Show path to stage directory Default: False

docs

Open buildtest docs in browser

```
buildtest docs [-h]
```

schemadocs

Open buildtest schema docs in browser

```
buildtest schemadocs [-h]
```

help (h)

buildtest command guide

```
buildtest help [-h] {build,buildspec,cdash,config,edit,history,inspect,path,report,  
↪ schema}
```

Positional Arguments

command	
	Possible choices: build, buildspec, cdash, config, edit, history, inspect, path, report, schema
	Show help message for command

GitHub: <https://github.com/buildtesters/buildtest> Documentation: <https://buildtest.readthedocs.io/en/latest/index.html> Schema Documentation: <https://buildtesters.github.io/buildtest/> Slack: <http://hpcbuildtest.slack.com/>

Please report issues at <https://github.com/buildtesters/buildtest/issues>

Copyright (c) 2021, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from the U.S. Dept. of Energy), Shahzeb Siddiqui, and Vanessa Sochat. All rights reserved.

LICENSE

buildtest is released under the [MIT license](#)

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

b

- `buildtest`, 442
- `buildtest.buildsystem`, 442
- `buildtest.buildsystem.base`, 442
- `buildtest.buildsystem.batch`, 447
- `buildtest.buildsystem.builders`, 448
- `buildtest.buildsystem.compilerbuilder`, 451
- `buildtest.buildsystem.parser`, 453
- `buildtest.buildsystem.scriptbuilder`, 454
- `buildtest.buildsystem.spack`, 455
- `buildtest.cli`, 456
- `buildtest.cli.build`, 456
- `buildtest.cli.buildspec`, 461
- `buildtest.cli.cd`, 467
- `buildtest.cli.cdash`, 467
- `buildtest.cli.clean`, 468
- `buildtest.cli.compilers`, 469
- `buildtest.cli.config`, 470
- `buildtest.cli.edit`, 472
- `buildtest.cli.help`, 472
- `buildtest.cli.history`, 473
- `buildtest.cli.inspect`, 474
- `buildtest.cli.path`, 476
- `buildtest.cli.report`, 477
- `buildtest.cli.schema`, 481
- `buildtest.config`, 508
- `buildtest.defaults`, 511
- `buildtest.exceptions`, 511
- `buildtest.executors`, 483
- `buildtest.executors.base`, 483
- `buildtest.executors.cobalt`, 484
- `buildtest.executors.job`, 486
- `buildtest.executors.local`, 487
- `buildtest.executors.lsf`, 488
- `buildtest.executors.pbs`, 491
- `buildtest.executors.poll`, 495
- `buildtest.executors.setup`, 495
- `buildtest.executors.slurm`, 497
- `buildtest.log`, 512
- `buildtest.main`, 513
- `buildtest.schemas`, 500
- `buildtest.schemas.defaults`, 500
- `buildtest.schemas.utils`, 501
- `buildtest.system`, 513
- `buildtest.utils`, 502
- `buildtest.utils.command`, 502
- `buildtest.utils.file`, 503
- `buildtest.utils.shell`, 506
- `buildtest.utils.timer`, 507
- `buildtest.utils.tools`, 507

Symbols

- `__enter__()` (*buildtest.utils.command.Capturing method*), 502
- `__exit__()` (*buildtest.utils.command.Capturing method*), 503
- `__missing__()` (*buildtest.utils.tools.Hasher method*), 508
- `__repr__()` (*buildtest.buildsystem.base.BuilderBase method*), 447
- `__repr__()` (*buildtest.buildsystem.parser.BuildspecParser method*), 453
- `__repr__()` (*buildtest.executors.base.BaseExecutor method*), 484
- `__repr__()` (*buildtest.executors.setup.BuildExecutor method*), 496
- `__repr__()` (*buildtest.utils.shell.Shell method*), 507
- `__str__()` (*buildtest.buildsystem.base.BuilderBase method*), 447
- `__str__()` (*buildtest.buildsystem.parser.BuildspecParser method*), 453
- `__str__()` (*buildtest.exceptions.BuildTestError method*), 511
- `__str__()` (*buildtest.exceptions.BuildspecError method*), 512
- `__str__()` (*buildtest.exceptions.ConfigurationError method*), 512
- `__str__()` (*buildtest.executors.base.BaseExecutor method*), 484
- `__str__()` (*buildtest.executors.setup.BuildExecutor method*), 496
- `__str__()` (*buildtest.utils.shell.Shell method*), 507
- `__str__()` (*buildtest.utils.tools.Hasher method*), 508
- `__version__` (in module *buildtest*), 516
- `_build_compilers()` (*buildtest.buildsystem.builders.Builder method*), 449
- `_build_setup()` (*buildtest.buildsystem.base.BuilderBase method*), 445
- `_check_executor()` (*buildtest.buildsystem.parser.BuildspecParser method*), 453
- `_check_filter_fields()` (*buildtest.cli.buildspec.BuildspecCache method*), 463
- `_check_filter_fields()` (*buildtest.cli.report.Report method*), 478
- `_check_format_fields()` (*buildtest.cli.buildspec.BuildspecCache method*), 463
- `_check_format_fields()` (*buildtest.cli.report.Report method*), 478
- `_check_regex()` (*buildtest.buildsystem.base.BuilderBase method*), 446
- `_check_runtime()` (*buildtest.buildsystem.base.BuilderBase method*), 446
- `_check_schema_type()` (*buildtest.buildsystem.parser.BuildspecParser method*), 453
- `_choose_executor()` (*buildtest.executors.setup.BuildExecutor method*), 496
- `_compile_cmd()` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder method*), 452
- `_default_test_variables()` (*buildtest.buildsystem.base.BuilderBase method*), 445
- `_detect_lang()` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder method*), 452
- `_discover_buildspecs()` (*buildtest.cli.buildspec.BuildspecCache method*), 463
- `_emit_command()` (*buildtest.buildsystem.base.BuilderBase method*), 445
- `_executor_check()` (*buildtest.config.SiteConfiguration method*), 509
- `_file` (*buildtest.config.SiteConfiguration attribute*), 508
- `_filter_buildspecs()` (*buildtest.cli.buildspec.BuildspecCache method*), 463
- `_filter_by_executor()` (*buildtest.cli.report.Report method*), 478
- `_filter_by_names()` (*buildtest.cli.report.Report method*), 478
- `_filter_by_returncode()` (*buildtest.cli.report.Report method*), 479
- `_filter_by_state()` (*buildtest.cli.report.Report method*), 479

<code>_filter_by_tags()</code>	(<i>buildtest.cli.report.Report</i> method), 478	<code>_skip_tests_by_tags()</code>	(<i>buildtest.buildsystem.builders.Builder</i> method), 449
<code>_generate_builders()</code>	(<i>buildtest.buildsystem.builders.Builder</i> method), 449	<code>_skip_tests_by_type()</code>	(<i>buildtest.buildsystem.builders.Builder</i> method), 450
<code>_generate_unique_id()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 443	<code>_skip_tests_run_only()</code>	(<i>buildtest.buildsystem.builders.Builder</i> method), 450
<code>_get_burst_buffer()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 445	<code>_spack_environment()</code>	(<i>buildtest.buildsystem.spack.SpackBuilder</i> method), 455
<code>_get_clusters()</code>	(<i>buildtest.system.Slurm</i> method), 514	<code>_testid_lookup()</code>	(<i>buildtest.cli.report.Report</i> method), 480
<code>_get_data_warp()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 446	<code>_update_build_history()</code>	(<i>buildtest.cli.build.BuildTest</i> method), 460
<code>_get_environment()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 446	<code>_update_compiler_section()</code>	(<i>buildtest.cli.compilers.BuildtestCompilers</i> method), 469
<code>_get_modules()</code>	(<i>buildtest.buildsystem.compilerbuilder.CompilerBuilder</i> method), 452	<code>_validate()</code>	(<i>buildtest.buildsystem.parser.BuildspecParser</i> method), 454
<code>_get_partitions()</code>	(<i>buildtest.system.Slurm</i> method), 514	<code>_validate_buildspecs()</code>	(<i>buildtest.cli.buildspec.BuildspecCache</i> method), 463
<code>_get_qos()</code>	(<i>buildtest.system.Slurm</i> method), 515	<code>_validate_cobalt_executors()</code>	(<i>buildtest.config.SiteConfiguration</i> method), 510
<code>_get_queues()</code>	(<i>buildtest.system.Cobalt</i> method), 515	<code>_validate_filters()</code>	(<i>buildtest.cli.build.BuildTest</i> method), 459
<code>_get_queues()</code>	(<i>buildtest.system.LSF</i> method), 515	<code>_validate_local_executors()</code>	(<i>buildtest.config.SiteConfiguration</i> method), 509
<code>_get_queues()</code>	(<i>buildtest.system.PBS</i> method), 516	<code>_validate_lsf_executors()</code>	(<i>buildtest.config.SiteConfiguration</i> method), 510
<code>_get_variables()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 446	<code>_validate_modules()</code>	(<i>buildtest.cli.compilers.BuildtestCompilers</i> method), 469
<code>_is_local_executor()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 443	<code>_validate_pbs_executors()</code>	(<i>buildtest.config.SiteConfiguration</i> method), 510
<code>_print_build_phase()</code>	(<i>buildtest.cli.build.BuildTest</i> method), 460	<code>_validate_slurm_executors()</code>	(<i>buildtest.config.SiteConfiguration</i> method), 510
<code>_print_run_phase()</code>	(<i>buildtest.cli.build.BuildTest</i> method), 460	<code>_write_build_script()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 445
<code>_print_test_summary()</code>	(<i>buildtest.cli.build.BuildTest</i> method), 460	<code>_write_buildspec_cache()</code>	(<i>buildtest.cli.buildspec.BuildspecCache</i> method), 463
<code>_process_compiler_config()</code>	(<i>buildtest.buildsystem.compilerbuilder.CompilerBuilder</i> method), 452	<code>_write_test()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 445
<code>_resolve_source()</code>	(<i>buildtest.buildsystem.compilerbuilder.CompilerBuilder</i> method), 452		
<code>_resolve_spack_root()</code>	(<i>buildtest.buildsystem.spack.SpackBuilder</i> method), 455		
<code>_returncode_check()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 446		
<code>_run_cmd()</code>	(<i>buildtest.buildsystem.compilerbuilder.CompilerBuilder</i> method), 452		
<code>_set_execute_perm()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 446		
<code>_set_metadata_values()</code>	(<i>buildtest.buildsystem.base.BuilderBase</i> method), 443		

A

`add_builder()` (*buildtest.executors.base.BaseExecutor* method), 483
`add_metrics()` (*buildtest.buildsystem.base.BuilderBase* method), 446

B

BaseExecutor (class in *buildtest.executors.base*), 483
BatchScript (class in *buildtest.buildsystem.batch*), 447
binaries (*buildtest.system.Cobalt* attribute), 515
binaries (*buildtest.system.LSF* attribute), 515
binaries (*buildtest.system.PBS* attribute), 516
binaries (*buildtest.system.Slurm* attribute), 514
`breakdown_by_test_names()` (*buildtest.cli.report.Report* method), 480
`build()` (*buildtest.buildsystem.base.BuilderBase* method), 444
`build()` (*buildtest.cli.build.BuildTest* method), 459
`build()` (*buildtest.cli.buildspec.BuildspecCache* method), 462
`build_cache()` (*buildtest.cli.buildspec.BuildspecCache* method), 463
`build_history()` (in module *buildtest.cli.history*), 474
BUILD_HISTORY_DIR (in module *buildtest.defaults*), 511
`build_menu()` (in module *buildtest.cli*), 482
`build_phase()` (*buildtest.cli.build.BuildTest* method), 459
BUILD_REPORT (in module *buildtest.cli*), 482
BUILD_REPORT (in module *buildtest.defaults*), 511
Builder (class in *buildtest.buildsystem.builders*), 448
`builder_names()` (*buildtest.cli.report.Report* method), 480
BuilderBase (class in *buildtest.buildsystem.base*), 442
BuildExecutor (class in *buildtest.executors.setup*), 496
buildspec (*buildtest.buildsystem.base.BuilderBase* attribute), 443
BUILDSPEC_CACHE_FILE (in module *buildtest.defaults*), 511
BUILDSPEC_DEFAULT_PATH (in module *buildtest.defaults*), 511
`buildspec_find()` (in module *buildtest.cli.buildspec*), 467
`buildspec_menu()` (in module *buildtest.cli*), 482
`buildspec_validate()` (in module *buildtest.cli.buildspec*), 466
BuildspecCache (class in *buildtest.cli.buildspec*), 462
BuildspecError, 511
BuildspecParser (class in *buildtest.buildsystem.parser*), 453
buildtest module, 442
BuildTest (class in *buildtest.cli.build*), 458
buildtest.buildsystem module, 442

buildtest.buildsystem.base module, 442
buildtest.buildsystem.batch module, 447
buildtest.buildsystem.builders module, 448
buildtest.buildsystem.compilerbuilder module, 451
buildtest.buildsystem.parser module, 453
buildtest.buildsystem.scriptbuilder module, 454
buildtest.buildsystem.spack module, 455
buildtest.cli module, 456
buildtest.cli.build module, 456
buildtest.cli.buildspec module, 461
buildtest.cli.cd module, 467
buildtest.cli.cdash module, 467
buildtest.cli.clean module, 468
buildtest.cli.compilers module, 469
buildtest.cli.config module, 470
buildtest.cli.edit module, 472
buildtest.cli.help module, 472
buildtest.cli.history module, 473
buildtest.cli.inspect module, 474
buildtest.cli.path module, 476
buildtest.cli.report module, 477
buildtest.cli.schema module, 481
buildtest.config module, 508
buildtest.defaults module, 511
buildtest.exceptions module, 511
buildtest.executors module, 483
buildtest.executors.base module, 483

- buildtest.executors.cobalt
 - module, 484
 - buildtest.executors.job
 - module, 486
 - buildtest.executors.local
 - module, 487
 - buildtest.executors.lsf
 - module, 488
 - buildtest.executors.pbs
 - module, 491
 - buildtest.executors.poll
 - module, 495
 - buildtest.executors.setup
 - module, 495
 - buildtest.executors.slurm
 - module, 497
 - buildtest.log
 - module, 512
 - buildtest.main
 - module, 513
 - buildtest.schemas
 - module, 500
 - buildtest.schemas.defaults
 - module, 500
 - buildtest.schemas.utils
 - module, 501
 - buildtest.system
 - module, 513
 - buildtest.utils
 - module, 502
 - buildtest.utils.command
 - module, 502
 - buildtest.utils.file
 - module, 503
 - buildtest.utils.shell
 - module, 506
 - buildtest.utils.timer
 - module, 507
 - buildtest.utils.tools
 - module, 507
 - BUILDTEST_BUILDSPEC_DIR (in module
 - buildtest.defaults), 511
 - BUILDTEST_COPYRIGHT (in module buildtest), 516
 - BUILDTEST_COPYRIGHT (in module buildtest.cli), 482
 - BUILDTEST_DEFAULT_TESTDIR (in module
 - buildtest.defaults), 511
 - BUILDTEST_EXECUTOR_DIR (in module
 - buildtest.defaults), 511
 - buildtest_help() (in module buildtest.cli.help), 473
 - BUILDTEST_REPORT_SUMMARY (in module
 - buildtest.defaults), 511
 - BUILDTEST_ROOT (in module buildtest.defaults), 511
 - BUILDTEST_USER_HOME (in module buildtest.defaults), 511
 - BUILDTEST_VERSION (in module buildtest), 516
 - BUILDTEST_VERSION (in module buildtest.cli), 482
 - BuildTestCommand (class in buildtest.utils.command), 503
 - BuildtestCompilers (class in buildtest.cli.compilers), 469
 - BuildTestError, 511
 - BuildTestSystem (class in buildtest.system), 514
- ## C
- cancel() (buildtest.executors.cobalt.CobaltJob
 - method), 486
 - cancel() (buildtest.executors.job.Job method), 486
 - cancel() (buildtest.executors.lsf.LSFJob method), 491
 - cancel() (buildtest.executors.pbs.PBSJob method), 494
 - cancel() (buildtest.executors.slurm.SlurmJob method), 499
 - cancelled() (buildtest.executors.poll.PollQueue
 - method), 495
 - Capturing (class in buildtest.utils.command), 502
 - cc (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 - attribute), 451
 - cdash_cmd() (in module buildtest.cli.cdash), 467
 - cdash_menu() (in module buildtest.cli), 483
 - cflags (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 - attribute), 451
 - change_directory() (in module buildtest.cli.cd), 467
 - check() (buildtest.executors.local.LocalExecutor
 - method), 487
 - check() (buildtest.system.BuildTestSystem method), 514
 - check() (buildtest.system.Scheduler method), 514
 - check_scheduler() (buildtest.system.BuildTestSystem
 - method), 514
 - check_test_state() (buildtest.buildsystem.base.BuilderBase
 - method), 447
 - clean() (in module buildtest.cli.clean), 468
 - cleanup() (buildtest.utils.command.Capturing method), 503
 - Cobalt (class in buildtest.system), 515
 - cobalt_log() (buildtest.executors.cobalt.CobaltJob
 - method), 485
 - CobaltBatchScript (class in
 - buildtest.buildsystem.batch), 448
 - CobaltExecutor (class in buildtest.executors.cobalt), 484
 - CobaltJob (class in buildtest.executors.cobalt), 485
 - compiler_cmd() (in module buildtest.cli.compilers), 469
 - compiler_find() (in module buildtest.cli.compilers), 469
 - compiler_table (buildtest.cli.compilers.BuildtestCompilers
 - attribute), 469
 - CompilerBuilder (class in
 - buildtest.buildsystem.compilerbuilder), 451

- `complete()` (*buildtest.executors.slurm.SlurmJob method*), 499
- `completed()` (*buildtest.executors.poll.PollQueue method*), 495
- `config` (*buildtest.config.SiteConfiguration attribute*), 509
- `config_cmd()` (*in module buildtest.cli.config*), 470
- `config_menu()` (*in module buildtest.cli*), 482
- `ConfigurationError`, 512
- `copy_stage_files()` (*buildtest.buildsystem.base.BuilderBase method*), 445
- `cppflags` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder attribute*), 451
- `create_dir()` (*in module buildtest.utils.file*), 504
- `custom_validator()` (*in module buildtest.schemas.defaults*), 501
- `cxx` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder attribute*), 451
- `cxxflags` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder attribute*), 451
- ## D
- `decode()` (*buildtest.utils.command.BuildTestCommand method*), 503
- `deep_get()` (*in module buildtest.utils.tools*), 508
- `default_format_fields` (*buildtest.cli.buildspec.BuildspecCache attribute*), 462
- `DEFAULT_SETTINGS_FILE` (*in module buildtest.defaults*), 511
- `DEFAULT_SETTINGS_SCHEMA` (*in module buildtest.defaults*), 511
- `detect_module_tool()` (*buildtest.system.BuildTestSystem method*), 514
- `detect_system()` (*buildtest.config.SiteConfiguration method*), 509
- `disabled_executors` (*buildtest.config.SiteConfiguration attribute*), 509
- `discover_buildspecs()` (*in module buildtest.cli.build*), 457
- `discover_buildspecs_by_executor()` (*in module buildtest.cli.build*), 457
- `discover_buildspecs_by_tags()` (*in module buildtest.cli.build*), 457
- `discover_by_buildspecs()` (*in module buildtest.cli.build*), 458
- `dispatch()` (*buildtest.executors.cobalt.CobaltExecutor method*), 485
- `dispatch()` (*buildtest.executors.lsf.LSFExecutor method*), 488
- `dispatch()` (*buildtest.executors.pbs.PBSExecutor method*), 492
- `dispatch()` (*buildtest.executors.slurm.SlurmExecutor method*), 498
- `display_table` (*buildtest.cli.report.Report attribute*), 478
- `duration()` (*buildtest.utils.timer.Timer method*), 507
- ## E
- `edit_buildspec()` (*in module buildtest.cli.edit*), 472
- `edit_menu()` (*in module buildtest.cli*), 482
- `endtime()` (*buildtest.buildsystem.base.BuilderBase method*), 444
- `err()` (*buildtest.utils.command.Capturing property*), 503
- `error()` (*buildtest.buildsystem.base.BuilderBase method*), 446
- `error_file()` (*buildtest.executors.cobalt.CobaltJob method*), 486
- `error_file()` (*buildtest.executors.lsf.LSFJob method*), 489
- `error_file()` (*buildtest.executors.pbs.PBSJob method*), 493
- `execute()` (*buildtest.utils.command.BuildTestCommand method*), 503
- `executor` (*buildtest.buildsystem.base.BuilderBase attribute*), 442
- `executor_breakdown()` (*buildtest.cli.buildspec.BuildspecCache method*), 464
- `ExecutorError`, 512
- `exitcode()` (*buildtest.executors.cobalt.CobaltJob method*), 486
- `exitcode()` (*buildtest.executors.lsf.LSFJob method*), 489
- `exitcode()` (*buildtest.executors.pbs.PBSJob method*), 493
- `exitcode()` (*buildtest.executors.slurm.SlurmJob method*), 499
- ## F
- `fail()` (*buildtest.executors.pbs.PBSJob method*), 493
- `failure()` (*buildtest.buildsystem.base.BuilderBase method*), 444
- `fc` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder attribute*), 451
- `fetch_records_by_ids()` (*buildtest.cli.report.Report method*), 480
- `fflags` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder attribute*), 451
- `file()` (*buildtest.config.SiteConfiguration property*), 509
- `FILE_LOG` (*in module buildtest.log*), 513
- `filter_buildspecs_from_report()` (*buildtest.cli.report.Report method*), 478
- `filter_fields` (*buildtest.cli.buildspec.BuildspecCache attribute*), 462
- `filter_fields` (*buildtest.cli.report.Report attribute*), 478

[find_buildspecs\(\)](#) (*buildtest.cli.buildspec.BuildspecCache* method), 463
[find_compilers\(\)](#) (*buildtest.cli.compilers.BuildtestCompilers* method), 469
[format_fields](#) (*buildtest.cli.buildspec.BuildspecCache* attribute), 462
[format_fields](#) (*buildtest.cli.report.Report* attribute), 478
G
[gather\(\)](#) (*buildtest.executors.cobalt.CobaltExecutor* method), 485
[gather\(\)](#) (*buildtest.executors.cobalt.CobaltJob* method), 486
[gather\(\)](#) (*buildtest.executors.lsf.LSFExecutor* method), 489
[gather\(\)](#) (*buildtest.executors.lsf.LSFJob* method), 490
[gather\(\)](#) (*buildtest.executors.pbs.PBSExecutor* method), 492
[gather\(\)](#) (*buildtest.executors.pbs.PBSJob* method), 494
[gather\(\)](#) (*buildtest.executors.slurm.SlurmExecutor* method), 498
[gather\(\)](#) (*buildtest.executors.slurm.SlurmJob* method), 499
[generate_script\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 446
[generate_script\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 451
[generate_script\(\)](#) (*buildtest.buildsystem.scriptbuilder.ScriptBuilder* method), 454
[generate_script\(\)](#) (*buildtest.buildsystem.spack.SpackBuilder* method), 455
[get\(\)](#) (*buildtest.cli.report.Report* method), 478
[get\(\)](#) (*buildtest.executors.job.Job* method), 486
[get\(\)](#) (*buildtest.executors.setup.BuildExecutor* method), 496
[get\(\)](#) (*buildtest.system.BuildTestSystem* method), 514
[get\(\)](#) (*buildtest.utils.shell.Shell* method), 507
[get\(\)](#) (*buildtest.utils.tools.Hasher* method), 508
[get_builder\(\)](#) (*buildtest.executors.base.BaseExecutor* method), 483
[get_builders\(\)](#) (*buildtest.buildsystem.builders.Builder* method), 450
[get_builders\(\)](#) (*buildtest.executors.setup.BuildExecutor* method), 497
[get_buildspecs\(\)](#) (*buildtest.cli.report.Report* method), 480
[get_cache\(\)](#) (*buildtest.cli.buildspec.BuildspecCache* method), 462
[get_cc\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_cflags\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_cobalt_directives\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 445
[get_command\(\)](#) (*buildtest.utils.command.BuildTestCommand* method), 503
[get_cppflags\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_cxx\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_cxxflags\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_error\(\)](#) (*buildtest.utils.command.BuildTestCommand* method), 503
[get_fc\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_fflags\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_headers\(\)](#) (*buildtest.buildsystem.batch.BatchScript* method), 447
[get_invalid_buildspecs\(\)](#) (*buildtest.cli.buildspec.BuildspecCache* method), 464
[get_job_directives\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 445
[get_ldflags\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_lsf_directives\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 445
[get_maintainers\(\)](#) (*buildtest.cli.buildspec.BuildspecCache* method), 464
[get_names\(\)](#) (*buildtest.cli.buildspec.BuildspecCache* method), 463
[get_names\(\)](#) (*buildtest.cli.report.Report* method), 480
[get_output\(\)](#) (*buildtest.utils.command.BuildTestCommand* method), 503
[get_parser\(\)](#) (in module *buildtest.cli*), 482
[get_path\(\)](#) (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* method), 452
[get_paths\(\)](#) (*buildtest.cli.buildspec.BuildspecCache* method), 464
[get_pbs_directives\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 445
[get_runtime\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 444
[get_slurm_directives\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 445
[get_test_extension\(\)](#) (*buildtest.buildsystem.base.BuilderBase* method), 443
[get_test_names\(\)](#) (*buildtest.buildsystem.builders.Builder*

method), 450
 get_testids() (*buildtest.cli.report.Report* method), 480
 get_unique_executors() (*buildtest.cli.buildspec.BuildspecCache* method), 464
 get_unique_tags() (*buildtest.cli.buildspec.BuildspecCache* method), 464
 get_valid_buildspecs() (*buildtest.cli.buildspec.BuildspecCache* method), 464

H

handle_kv_string() (*in module buildtest.cli*), 482
 Hasher (*class in buildtest.utils.tools*), 508
 here (*in module buildtest.schemas.defaults*), 501
 here (*in module buildtest.schemas.utils*), 502
 history_menu() (*in module buildtest.cli*), 482

I

init_logfile() (*in module buildtest.log*), 513
 inspect_buildspec() (*in module buildtest.cli.inspect*), 475
 inspect_by_id() (*in module buildtest.cli.inspect*), 476
 inspect_by_name() (*in module buildtest.cli.inspect*), 475
 inspect_cmd() (*in module buildtest.cli.inspect*), 475
 inspect_list() (*in module buildtest.cli.inspect*), 475
 inspect_menu() (*in module buildtest.cli*), 483
 inspect_query() (*in module buildtest.cli.inspect*), 475
 invalid_executors (*buildtest.config.SiteConfiguration* attribute), 509
 is_batch_job() (*buildtest.buildsystem.base.BuilderBase* method), 444
 is_cancelled() (*buildtest.executors.cobalt.CobaltJob* method), 485
 is_cancelled() (*buildtest.executors.slurm.SlurmJob* method), 498
 is_cobalt() (*buildtest.executors.setup.BuildExecutor* method), 496
 is_complete() (*buildtest.buildsystem.base.BuilderBase* method), 444
 is_complete() (*buildtest.executors.cobalt.CobaltJob* method), 485
 is_complete() (*buildtest.executors.lsf.LSFJob* method), 489
 is_complete() (*buildtest.executors.pbs.PBSJob* method), 493
 is_complete() (*buildtest.executors.slurm.SlurmJob* method), 498
 is_dir() (*in module buildtest.utils.file*), 504
 is_failed() (*buildtest.executors.lsf.LSFJob* method), 489

is_failed() (*buildtest.executors.slurm.SlurmJob* method), 498
 is_failure() (*buildtest.buildsystem.base.BuilderBase* method), 444
 is_file() (*in module buildtest.utils.file*), 504
 is_int() (*in module buildtest.cli.report*), 477
 is_local() (*buildtest.executors.setup.BuildExecutor* method), 496
 is_lsf() (*buildtest.executors.setup.BuildExecutor* method), 496
 is_out_of_memory() (*buildtest.executors.slurm.SlurmJob* method), 499
 is_pbs() (*buildtest.executors.setup.BuildExecutor* method), 496
 is_pending() (*buildtest.executors.cobalt.CobaltJob* method), 485
 is_pending() (*buildtest.executors.job.Job* method), 486
 is_pending() (*buildtest.executors.lsf.LSFJob* method), 489
 is_pending() (*buildtest.executors.pbs.PBSJob* method), 493
 is_pending() (*buildtest.executors.slurm.SlurmJob* method), 498
 is_running() (*buildtest.executors.cobalt.CobaltJob* method), 485
 is_running() (*buildtest.executors.job.Job* method), 486
 is_running() (*buildtest.executors.lsf.LSFJob* method), 489
 is_running() (*buildtest.executors.pbs.PBSJob* method), 493
 is_running() (*buildtest.executors.slurm.SlurmJob* method), 498
 is_slurm() (*buildtest.executors.setup.BuildExecutor* method), 496
 is_suspended() (*buildtest.executors.cobalt.CobaltJob* method), 485
 is_suspended() (*buildtest.executors.job.Job* method), 486
 is_suspended() (*buildtest.executors.lsf.LSFJob* method), 489
 is_suspended() (*buildtest.executors.pbs.PBSJob* method), 493
 is_suspended() (*buildtest.executors.slurm.SlurmJob* method), 498
 is_timeout() (*buildtest.executors.slurm.SlurmJob* method), 499
 is_unknown() (*buildtest.buildsystem.base.BuilderBase* method), 444

J

job (*buildtest.buildsystem.base.BuilderBase* attribute), 442
 Job (*class in buildtest.executors.job*), 486

L

- `lang_ext_table` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* attribute), 451
- `latest_testid_by_name()` (*buildtest.cli.report.Report* method), 480
- `launcher_command()` (*buildtest.executors.cobalt.CobaltExecutor* method), 485
- `launcher_command()` (*buildtest.executors.lsf.LSFExecutor* method), 488
- `launcher_command()` (*buildtest.executors.pbs.PBSExecutor* method), 492
- `launcher_command()` (*buildtest.executors.slurm.SlurmExecutor* method), 498
- `ldflags` (*buildtest.buildsystem.compilerbuilder.CompilerBuilder* attribute), 451
- `list()` (*buildtest.cli.compilers.BuildtestCompilers* method), 470
- `list_builds()` (in module *buildtest.cli.history*), 474
- `list_executors()` (*buildtest.executors.setup.BuildExecutor* method), 496
- `load()` (*buildtest.cli.report.Report* method), 478
- `load()` (*buildtest.config.SiteConfiguration* method), 509
- `load()` (*buildtest.executors.base.BaseExecutor* method), 483
- `load()` (*buildtest.executors.cobalt.CobaltExecutor* method), 485
- `load()` (*buildtest.executors.local.LocalExecutor* method), 487
- `load()` (*buildtest.executors.lsf.LSFExecutor* method), 488
- `load()` (*buildtest.executors.pbs.PBSExecutor* method), 492
- `load()` (*buildtest.executors.slurm.SlurmExecutor* method), 498
- `load_builders()` (*buildtest.executors.setup.BuildExecutor* method), 496
- `load_json()` (in module *buildtest.utils.file*), 506
- `load_paths()` (*buildtest.cli.buildspec.BuildspecCache* method), 462
- `load_recipe()` (in module *buildtest.schemas.utils*), 502
- `load_schema()` (in module *buildtest.schemas.utils*), 502
- `LocalExecutor` (class in *buildtest.executors.local*), 487
- `LOG_FORMATTER` (in module *buildtest.log*), 512
- `LOG_NAME` (in module *buildtest.log*), 512
- `logger` (*buildtest.system.Scheduler* attribute), 514
- `logger` (in module *buildtest.cli.build*), 456
- `logger` (in module *buildtest.cli.buildspec*), 462
- `logger` (in module *buildtest.cli.history*), 474
- `logger` (in module *buildtest.cli.report*), 477
- `logger` (in module *buildtest.config*), 508
- `logger` (in module *buildtest.executors.cobalt*), 484
- `logger` (in module *buildtest.executors.lsf*), 488
- `logger` (in module *buildtest.executors.pbs*), 491
- `logger` (in module *buildtest.executors.setup*), 496
- `logger` (in module *buildtest.executors.slurm*), 497
- `logger` (in module *buildtest.executors.spack*), 497
- `lookup_buildspec_by_name()` (*buildtest.cli.buildspec.BuildspecCache* method), 463
- `LSF` (class in *buildtest.system*), 515
- `LSFBatchScript` (class in *buildtest.buildsystem.batch*), 447
- `LSFExecutor` (class in *buildtest.executors.lsf*), 488
- `LSFJob` (class in *buildtest.executors.lsf*), 489

M

- `main()` (in module *buildtest.main*), 513
- `metadata` (*buildtest.buildsystem.base.BuilderBase* attribute), 442
- `module`
 - buildtest*, 442
 - buildtest.buildsystem*, 442
 - buildtest.buildsystem.base*, 442
 - buildtest.buildsystem.batch*, 447
 - buildtest.buildsystem.builders*, 448
 - buildtest.buildsystem.compilerbuilder*, 451
 - buildtest.buildsystem.parser*, 453
 - buildtest.buildsystem.scripbuilder*, 454
 - buildtest.buildsystem.spack*, 455
 - buildtest.cli*, 456
 - buildtest.cli.build*, 456
 - buildtest.cli.buildspec*, 461
 - buildtest.cli.cd*, 467
 - buildtest.cli.cdash*, 467
 - buildtest.cli.clean*, 468
 - buildtest.cli.compilers*, 469
 - buildtest.cli.config*, 470
 - buildtest.cli.edit*, 472
 - buildtest.cli.help*, 472
 - buildtest.cli.history*, 473
 - buildtest.cli.inspect*, 474
 - buildtest.cli.path*, 476
 - buildtest.cli.report*, 477
 - buildtest.cli.schema*, 481
 - buildtest.config*, 508
 - buildtest.defaults*, 511
 - buildtest.exceptions*, 511
 - buildtest.executors*, 483
 - buildtest.executors.base*, 483
 - buildtest.executors.cobalt*, 484
 - buildtest.executors.job*, 486
 - buildtest.executors.local*, 487
 - buildtest.executors.lsf*, 488
 - buildtest.executors.pbs*, 491
 - buildtest.executors.poll*, 495
 - buildtest.executors.setup*, 495
 - buildtest.executors.slurm*, 497
 - buildtest.log*, 512

buildtest.main, 513
 buildtest.schemas, 500
 buildtest.schemas.defaults, 500
 buildtest.schemas.utils, 501
 buildtest.system, 513
 buildtest.utils, 502
 buildtest.utils.command, 502
 buildtest.utils.file, 503
 buildtest.utils.shell, 506
 buildtest.utils.timer, 507
 buildtest.utils.tools, 507

N

name (*buildtest.buildsystem.base.BuilderBase* attribute), 442
 name() (*buildtest.config.SiteConfiguration* method), 509

O

opts() (*buildtest.utils.shell.Shell* property), 506
 out() (*buildtest.utils.command.Capturing* property), 503
 output() (*buildtest.buildsystem.base.BuilderBase* method), 446
 output_file() (*buildtest.executors.cobalt.CobaltJob* method), 485
 output_file() (*buildtest.executors.lsf.LSFJob* method), 489
 output_file() (*buildtest.executors.pbs.PBSJob* method), 493

P

parse_buildspecs() (*buildtest.cli.build.BuildTest* method), 459
 path() (*buildtest.utils.shell.Shell* property), 506
 path_cmd() (in module *buildtest.cli.path*), 476
 PBS (class in *buildtest.system*), 515
 PBSBatchScript (class in *buildtest.buildsystem.batch*), 448
 PBSExecutor (class in *buildtest.executors.pbs*), 491
 PBSJob (class in *buildtest.executors.pbs*), 492
 poll() (*buildtest.executors.cobalt.CobaltExecutor* method), 485
 poll() (*buildtest.executors.cobalt.CobaltJob* method), 486
 poll() (*buildtest.executors.job.Job* method), 487
 poll() (*buildtest.executors.lsf.LSFExecutor* method), 489
 poll() (*buildtest.executors.lsf.LSFJob* method), 489
 poll() (*buildtest.executors.pbs.PBSExecutor* method), 492
 poll() (*buildtest.executors.pbs.PBSJob* method), 493
 poll() (*buildtest.executors.poll.PollQueue* method), 495
 poll() (*buildtest.executors.slurm.SlurmExecutor* method), 498

poll() (*buildtest.executors.slurm.SlurmJob* method), 499
 poll_cmd() (*buildtest.executors.pbs.PBSExecutor* attribute), 492
 poll_phase() (*buildtest.cli.build.BuildTest* method), 460
 PollQueue (class in *buildtest.executors.poll*), 495
 positive_number() (in module *buildtest.cli*), 482
 post_run_steps() (*buildtest.buildsystem.base.BuilderBase* method), 446
 print_build_help() (in module *buildtest.cli.help*), 472
 print_buildspec_help() (in module *buildtest.cli.help*), 473
 print_buildspecfiles() (*buildtest.cli.buildspec.BuildspecCache* method), 464
 print_buildspecs() (*buildtest.cli.buildspec.BuildspecCache* method), 465
 print_by_executors() (*buildtest.cli.buildspec.BuildspecCache* method), 465
 print_by_tags() (*buildtest.cli.buildspec.BuildspecCache* method), 465
 print_cdash_help() (in module *buildtest.cli.help*), 473
 print_compilers() (*buildtest.cli.compilers.BuildtestCompilers* method), 470
 print_config_help() (in module *buildtest.cli.help*), 473
 print_discovered_buildspecs() (in module *buildtest.cli.build*), 457
 print_edit_help() (in module *buildtest.cli.help*), 473
 print_executors() (*buildtest.cli.buildspec.BuildspecCache* method), 464
 print_filter_fields() (*buildtest.cli.buildspec.BuildspecCache* static method), 466
 print_filter_fields() (*buildtest.cli.report.Report* method), 479
 print_filters() (in module *buildtest.cli.build*), 458
 print_format_fields() (*buildtest.cli.buildspec.BuildspecCache* static method), 466
 print_format_fields() (*buildtest.cli.report.Report* method), 479
 print_history_help() (in module *buildtest.cli.help*), 473
 print_inspect_help() (in module *buildtest.cli.help*), 473
 print_invalid_buildspecs() (*buildtest.cli.buildspec.BuildspecCache* method), 466
 print_json() (*buildtest.cli.compilers.BuildtestCompilers* method), 470
 print_maintainer() (*buildtest.cli.buildspec.BuildspecCache*

method), 465
 print_maintainers_by_buildspecs()
 (buildtest.cli.buildspec.BuildspecCache
 method), 465
 print_path_help() (in module buildtest.cli.help), 473
 print_paths() (buildtest.cli.buildspec.BuildspecCache
 method), 466
 print_pending_jobs()
 (buildtest.executors.poll.PollQueue method),
 495
 print_polled_jobs()
 (buildtest.executors.poll.PollQueue method),
 495
 print_report() (buildtest.cli.report.Report method),
 479
 print_report_help() (in module buildtest.cli.help),
 473
 print_schema_help() (in module buildtest.cli.help),
 473
 print_tags() (buildtest.cli.buildspec.BuildspecCache
 method), 464
 print_yaml() (buildtest.cli.compilers.BuildtestCompilers
 method), 470
 process_report() (buildtest.cli.report.Report
 method), 479

Q

query_builds() (in module buildtest.cli.history), 474

R

read_file() (in module buildtest.utils.file), 505
 recipe (buildtest.buildsystem.base.BuilderBase at-
 tribute), 442
 remove_file() (in module buildtest.utils.file), 506
 Report (class in buildtest.cli.report), 477
 report_cmd() (in module buildtest.cli.report), 480
 report_menu() (in module buildtest.cli), 483
 report_summary() (in module buildtest.cli.report), 480
 reportfile() (buildtest.cli.report.Report method), 478
 resolve() (buildtest.config.SiteConfiguration method),
 509
 resolve_path() (in module buildtest.utils.file), 505
 resolve_testdirectory() (in module
 buildtest.cli.build), 456
 resolver (in module buildtest.schemas.defaults), 501
 retry() (buildtest.buildsystem.base.BuilderBase
 method), 444
 returncode() (buildtest.utils.command.BuildTestCommand
 method), 503
 run() (buildtest.buildsystem.base.BuilderBase method),
 444
 run() (buildtest.executors.base.BaseExecutor method),
 483

run() (buildtest.executors.local.LocalExecutor method),
 487
 run() (buildtest.executors.setup.BuildExecutor method),
 497
 run_command() (buildtest.buildsystem.base.BuilderBase
 method), 444
 run_phase() (buildtest.cli.build.BuildTest method), 459
 runtime() (buildtest.buildsystem.base.BuilderBase
 method), 444
 RuntimeFailure, 512

S

sched_init() (buildtest.buildsystem.base.BuilderBase
 method), 445
 Scheduler (class in buildtest.system), 514
 schema_cmd() (in module buildtest.cli.schema), 481
 schema_menu() (in module buildtest.cli), 483
 SCHEMA_ROOT (in module buildtest.defaults), 511
 schema_store (in module buildtest.schemas.defaults),
 501
 schema_table (in module buildtest.cli), 482
 schema_table (in module buildtest.schemas.defaults),
 501
 ScriptBuilder (class in
 buildtest.buildsystem.scriptbuilder), 454
 set_cc() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_cflags() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_command() (buildtest.utils.command.BuildTestCommand
 method), 503
 set_cppflags() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_cxx() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_cxxflags() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_fc() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_fflags() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_ldflags() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 452
 set_stderr() (buildtest.utils.command.Capturing
 method), 503
 set_stdout() (buildtest.utils.command.Capturing
 method), 503
 setup() (buildtest.buildsystem.compilerbuilder.CompilerBuilder
 method), 451
 setup() (buildtest.executors.setup.BuildExecutor
 method), 496
 shebang (buildtest.buildsystem.base.BuilderBase at-
 tribute), 443
 Shell (class in buildtest.utils.shell), 506

- `show_buildspecs()` (in module `buildtest.cli.buildspec`), 466
- `single_kv_string()` (in module `buildtest.cli`), 482
- `SiteConfiguration` (class in `buildtest.config`), 508
- `sleep()` (`buildtest.executors.poll.PollQueue` method), 495
- `Slurm` (class in `buildtest.system`), 514
- `SlurmBatchScript` (class in `buildtest.buildsystem.batch`), 447
- `SlurmExecutor` (class in `buildtest.executors.slurm`), 497
- `SlurmJob` (class in `buildtest.executors.slurm`), 498
- `sorted_alphanumeric()` (in module `buildtest.cli.history`), 474
- `SpackBuilder` (class in `buildtest.buildsystem.spack`), 455
- `start()` (`buildtest.buildsystem.base.BuilderBase` method), 444
- `start()` (`buildtest.utils.timer.Timer` method), 507
- `starttime()` (`buildtest.buildsystem.base.BuilderBase` method), 444
- `state` (`buildtest.buildsystem.base.BuilderBase` attribute), 443
- `state()` (`buildtest.executors.job.Job` method), 486
- `state()` (`buildtest.executors.slurm.SlurmJob` method), 499
- `stop()` (`buildtest.buildsystem.base.BuilderBase` method), 444
- `stop()` (`buildtest.utils.timer.Timer` method), 507
- `success()` (`buildtest.buildsystem.base.BuilderBase` method), 444
- `success()` (`buildtest.executors.pbs.PBSJob` method), 493
- `summarize_buildspec_cache()` (in module `buildtest.cli.buildspec`), 466
- `supported_schemas` (in module `buildtest.defaults`), 511
- `supported_type_schemas` (in module `buildtest.defaults`), 511
- `system` (`buildtest.system.BuildTestSystem` attribute), 514
- `system` (in module `buildtest.system`), 516
- ## T
- `table` (`buildtest.cli.buildspec.BuildspecCache` attribute), 462
- `tag_breakdown()` (`buildtest.cli.buildspec.BuildspecCache` method), 464
- `target_config` (`buildtest.config.SiteConfiguration` attribute), 509
- `test_breakdown_by_buildspec()` (`buildtest.cli.buildspec.BuildspecCache` method), 464
- `testdir` (`buildtest.buildsystem.base.BuilderBase` attribute), 443
- `Timer` (class in `buildtest.utils.timer`), 507
- `TimerError`, 507
- `type` (`buildtest.buildsystem.compilerbuilder.CompilerBuilder` attribute), 451
- `type` (`buildtest.buildsystem.scriptbuilder.ScriptBuilder` attribute), 454
- `type` (`buildtest.buildsystem.spack.SpackBuilder` attribute), 455
- `type` (`buildtest.executors.base.BaseExecutor` attribute), 483
- `type` (`buildtest.executors.cobalt.CobaltExecutor` attribute), 484
- `type` (`buildtest.executors.local.LocalExecutor` attribute), 487
- `type` (`buildtest.executors.lsf.LSFExecutor` attribute), 488
- `type` (`buildtest.executors.pbs.PBSExecutor` attribute), 492
- `type` (`buildtest.executors.slurm.SlurmExecutor` attribute), 498
- ## U
- `update_report()` (in module `buildtest.cli.build`), 461
- `upload_test_cdash()` (in module `buildtest.cli.cdash`), 468
- `USER_SETTINGS_FILE` (in module `buildtest.defaults`), 511
- `userhome` (in module `buildtest.defaults`), 511
- ## V
- `valid_executors` (`buildtest.config.SiteConfiguration` attribute), 509
- `valid_shells` (`buildtest.utils.shell.Shell` attribute), 506
- `validate()` (`buildtest.config.SiteConfiguration` method), 509
- `validate_config()` (in module `buildtest.cli.config`), 470
- `VAR_DIR` (in module `buildtest.defaults`), 511
- `view_configuration()` (in module `buildtest.cli.config`), 471
- `view_executors()` (in module `buildtest.cli.config`), 471
- `view_summary()` (in module `buildtest.cli.config`), 471
- `view_system()` (in module `buildtest.cli.config`), 470
- ## W
- `walk_tree()` (in module `buildtest.utils.file`), 504
- `workdir()` (`buildtest.executors.slurm.SlurmJob` method), 499
- `write_file()` (in module `buildtest.utils.file`), 505
- `write_python_script()` (`buildtest.buildsystem.scriptbuilder.ScriptBuilder` method), 454