Los Alamos
NATIONAL LABORATORY

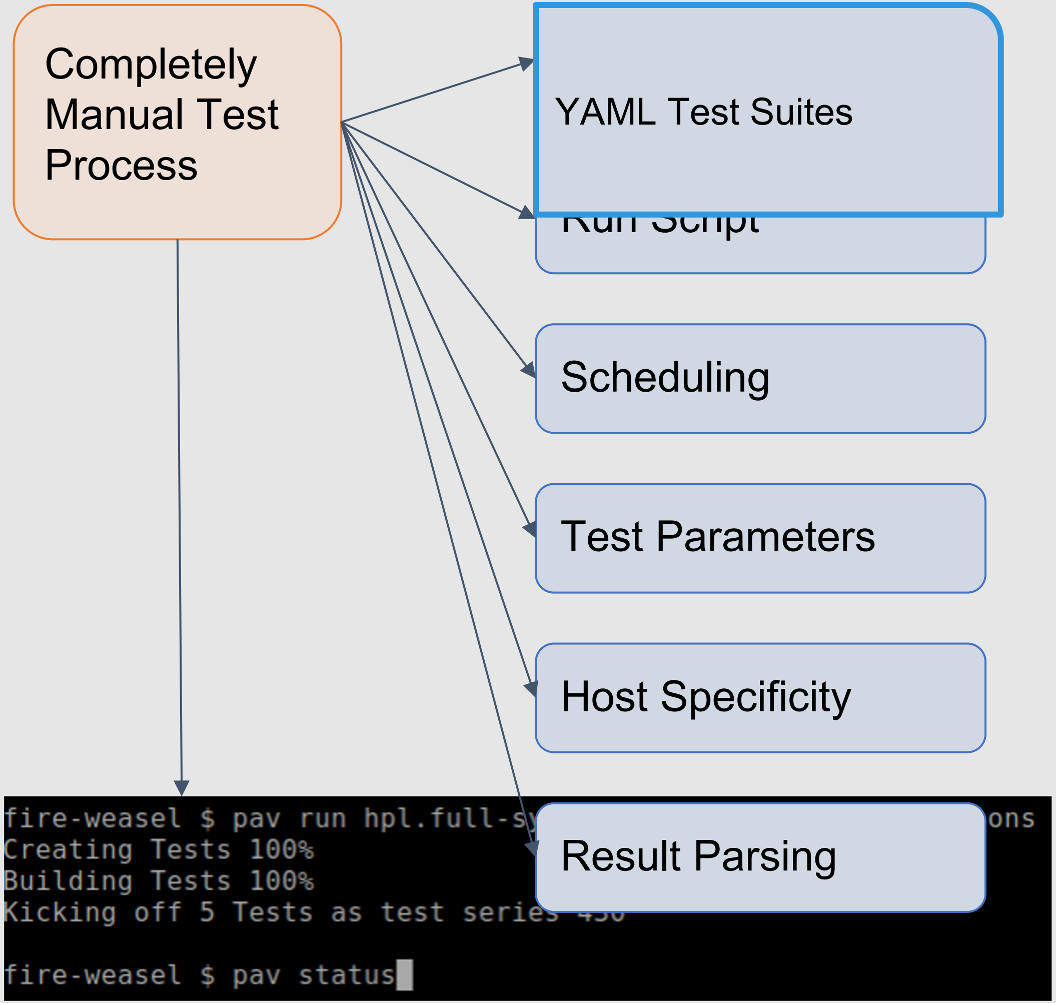
HPC Testing using Pavilion 2

Francine Lapid and Paul Ferrell
HPC-Environments Programming and Runtime Environments Team

HPC System Test Birds of a Feather at SC21
Tuesday, 16 November 2021, 5:15pm - 6:45pm CST, Rm 229
The International Conference for High Performance Computing, Networking, Storage, and Analysis

LA-UR-21-31303

Test Abstraction



Completely Manual Test Process

YAML Test Suites

Run Script

Scheduling

Test Parameters

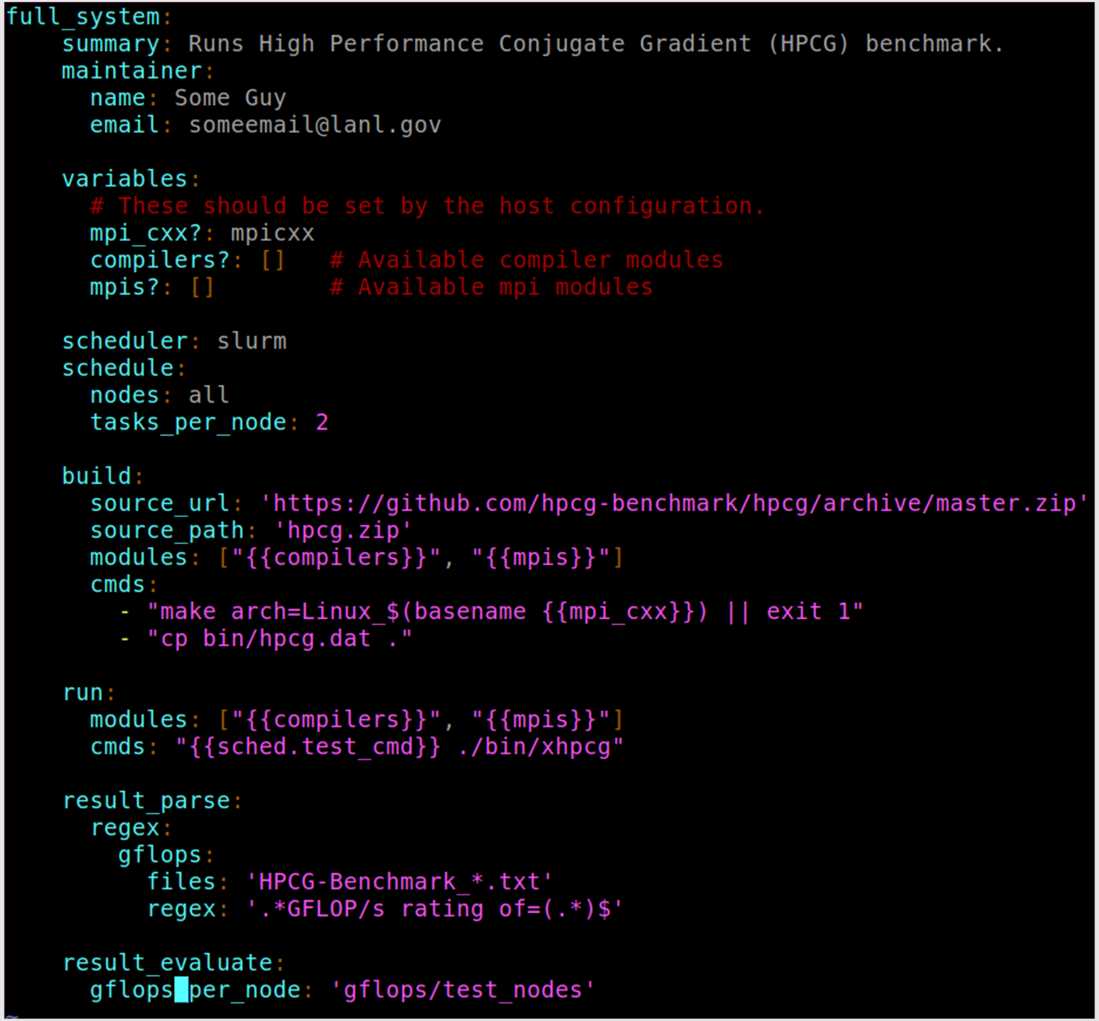
Host Specificity

Result Parsing

```
fire-weasel $ pav run hpl.full-sy                                   ons
Creating Tests 100%
Building Tests 100%
Kicking off 5 Tests as test series 430

fire-weasel $ pav status
```

Problems

- Capture Tester Knowledge
- Cluster Variance
- Results in Random Formats
- Different Schedulers
- Different Software Environments



```
full_system:
    summary: Runs High Performance Conjugate Gradient (HPCG) benchmark.
    maintainer:
      name: Some Guy
      email: someemail@lanl.gov

    variables:
      # These should be set by the host configuration.
      mpi_cxx?: mpicxx
      compilers?: []    # Available compiler modules
      mpis?: []         # Available mpi modules

    scheduler: slurm
    schedule:
      nodes: all
      tasks_per_node: 2

    build:
      source_url: 'https://github.com/hpcg-benchmark/hpcg/archive/master.zip'
      source_path: 'hpcg.zip'
      modules: ["{{compilers}}", "{{mpis}}"]
      cmds:
        - "make arch=Linux_$(basename {{mpi_cxx}}) || exit 1"
        - "cp bin/hpcg.dat ."

    run:
      modules: ["{{compilers}}", "{{mpis}}"]
      cmds: "{{sched.test_cmd}} ./bin/xhpcg"

    result_parse:
      regex:
        gflops:
          files: 'HPCG-Benchmark_*.txt'
          regex: '.*GFLOP/s rating of=(.*)$'

    result_evaluate:
      gflops_per_node: 'gflops/test_nodes'
```

Test Suite Features

Test Suite Features

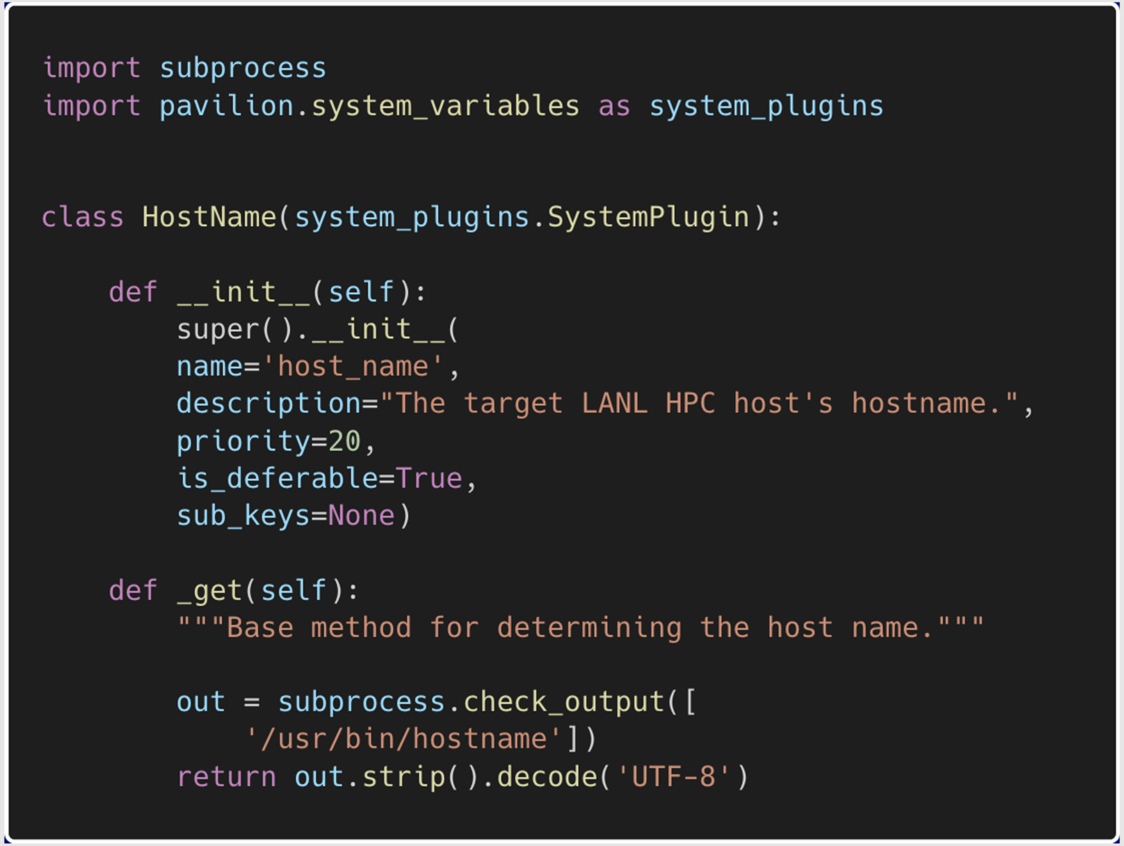
- Variables
- Host Dependent Variables
- Inheritance
- Module Abstractions
- Permutations
- Automatic Result Parsing
- Test Input File Creation
- and much more!

Plugins

Add plugins that let you customize Pavilion meet your testing needs.

Types of plugins:

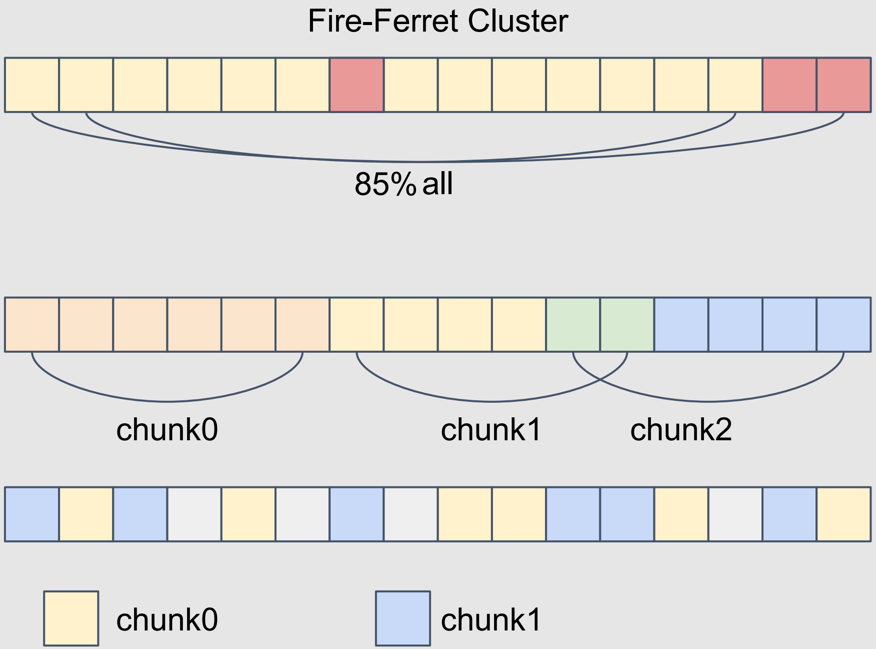
- Commands
- Expressions
- Module Wrappers
- Result Parsers
- Schedulers
- System Variables



```
import subprocess
import pavilion.system_variables as system_plugins


class HostName(system_plugins.SystemPlugin):

    def __init__(self):
        super().__init__(
        name='host_name',
        description="The target LANL HPC host's hostname.",
        priority=20,
        is_deferable=True,
        sub_keys=None)

    def _get(self):
        """Base method for determining the host name."""

        out = subprocess.check_output([
            '/usr/bin/hostname'])
        return out.strip().decode('UTF-8')
```

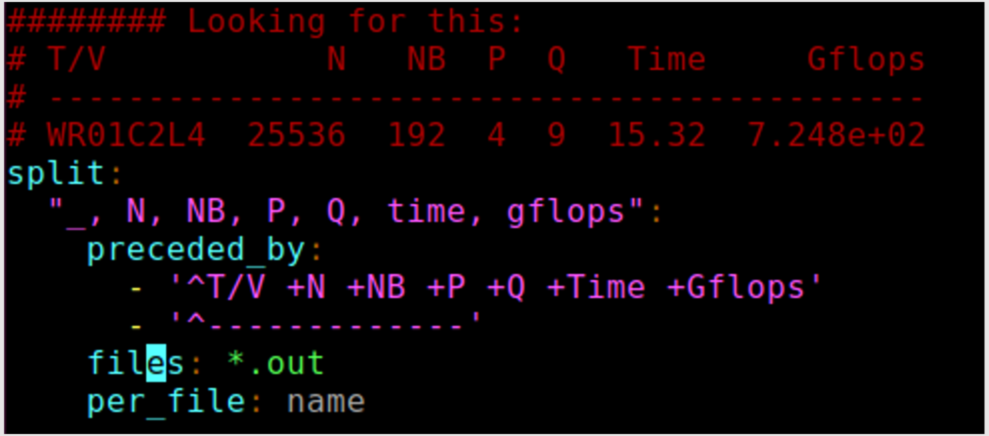
Scheduling

- Slurm
- Slurm w/mpirun
- Local
- LSF (under development)

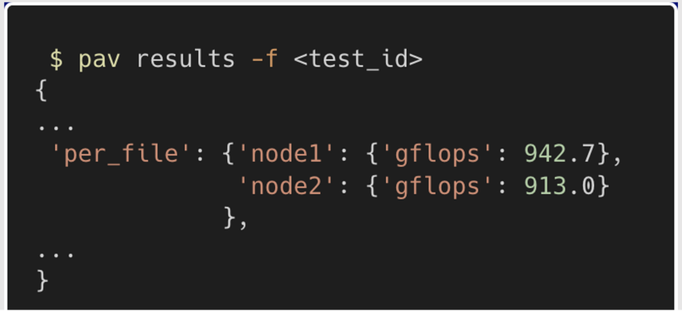


Data Capture

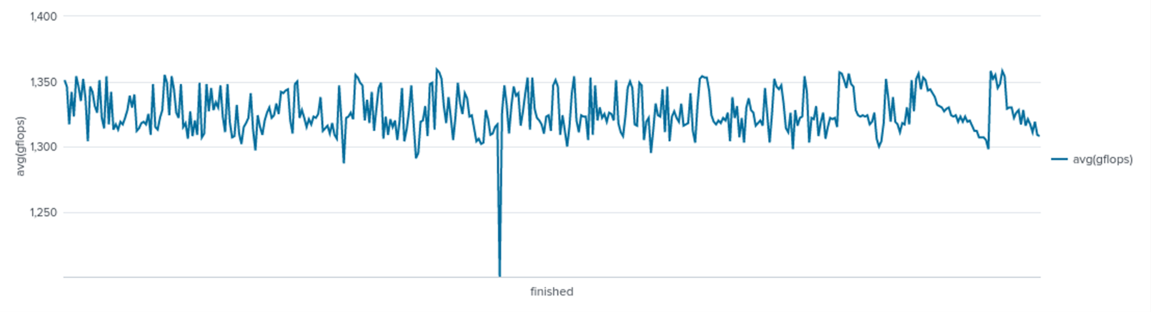
- Output gets parsed into JSON using user-set parameters
- Can use Splunk or pav graph
- Logs & scripts get stored for easy access



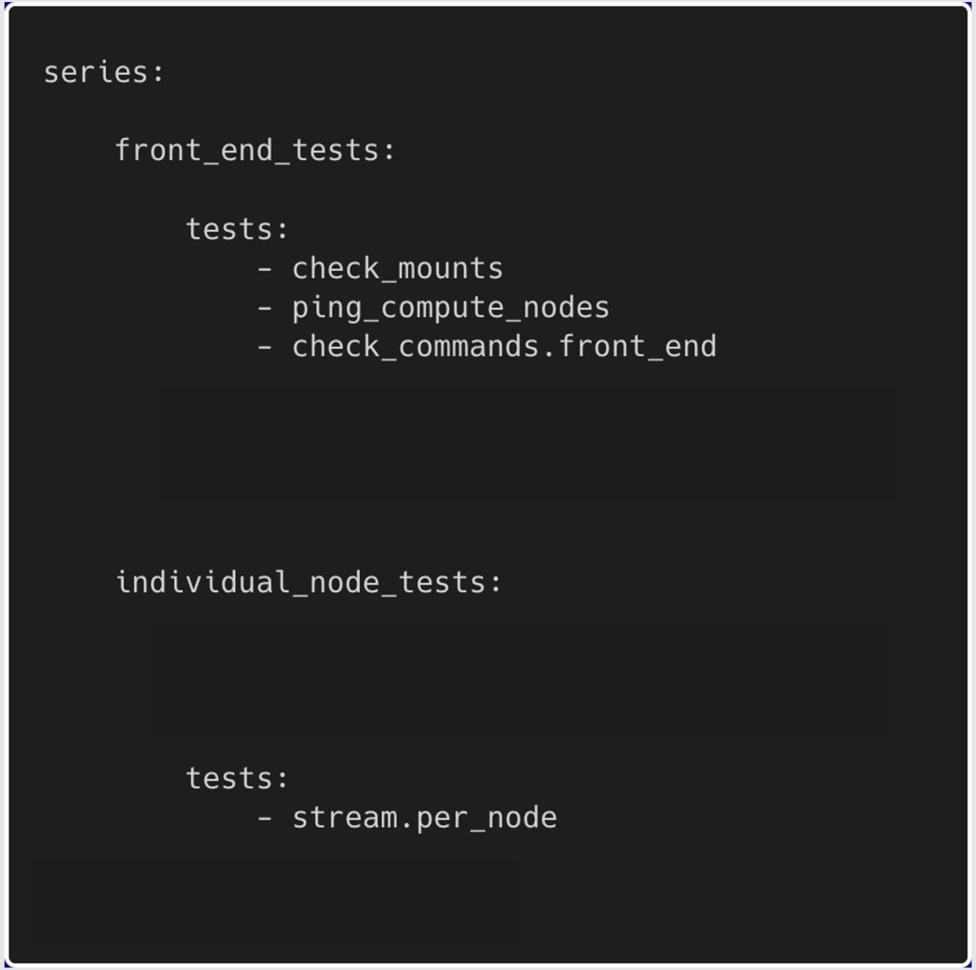
```
######## Looking for this:
# T/V            N    NB   P   Q    Time      Gflops
# ----------------------------------------------------
# WR01C2L4  25536   192   4   9   15.32   7.248e+02
split:
  "_, N, NB, P, Q, time, gflops":
    preceded_by:
      - '^T/V +N +NB +P +Q +Time +Gflops'
      - '^-------------'
    files: *.out
    per_file: name
```



```
 $ pav results -f <test_id>
{
...
 'per_file': {'node1': {'gflops': 942.7},
              'node2': {'gflops': 913.0}
             },
...
}
```



Test Planning



```
series:

    front_end_tests:

        tests:
            - check_mounts
            - ping_compute_nodes
            - check_commands.front_end



    individual_node_tests:



        tests:
            - stream.per_node
```

- pav series series_name
- specify test hierarchies
- set conditions that determine if a set of tests will run
- kickoff a series to run multiple times

Regression Testing

Procedure

- pav series regressions
- pav status
- investigate any problems with admins

Some tests we run

- hello_mpi
- hpcg
- hpl
- imb
- ior
- kickstart
- license-check
- mounts
- nhc
- slow_test
- stream
- supermagic
- ...

Conclusion

Pavilion is at:

https://github.com/hpc/pavilion2

Documentation is at:

https://pavilion2.readthedocs.io/en/latest/

pferrell@lanl.gov

lapid@lanl.gov