CI Updates

Nightly Builds

Mihai Patrascoiu
CERN / IT- ST- PDS

CI – setup overview

- Build for different platforms
- Code-related tasks (e.g.: static analysis)

- Create Docker image with EOS installed

- Testing infrastructure (docker + k8s)
- Publish RPMs

EOS Workshop 2018 – *New CI Platform for EOS and XrootD*
EOS Workshop 2019 – *EOS Testing Service development: leveraging CI + Kubernetes*

CI – 2018 and now

# of jobs	Build	Docker build	Testing	Publish
2018	6	2	4	5
2020	11	6	14	5

- Execution time: 40m → 1h (or more)

Reflections on the CI

- Clear tendency of pipeline functionality to grow
- Execution time increases
- Pipeline may become congested (e.g.: many jobs, not enough runners)
- Runners reach dreaded OOM/timeout (e.g.: 2 build jobs on the same runner)

Nightly Builds

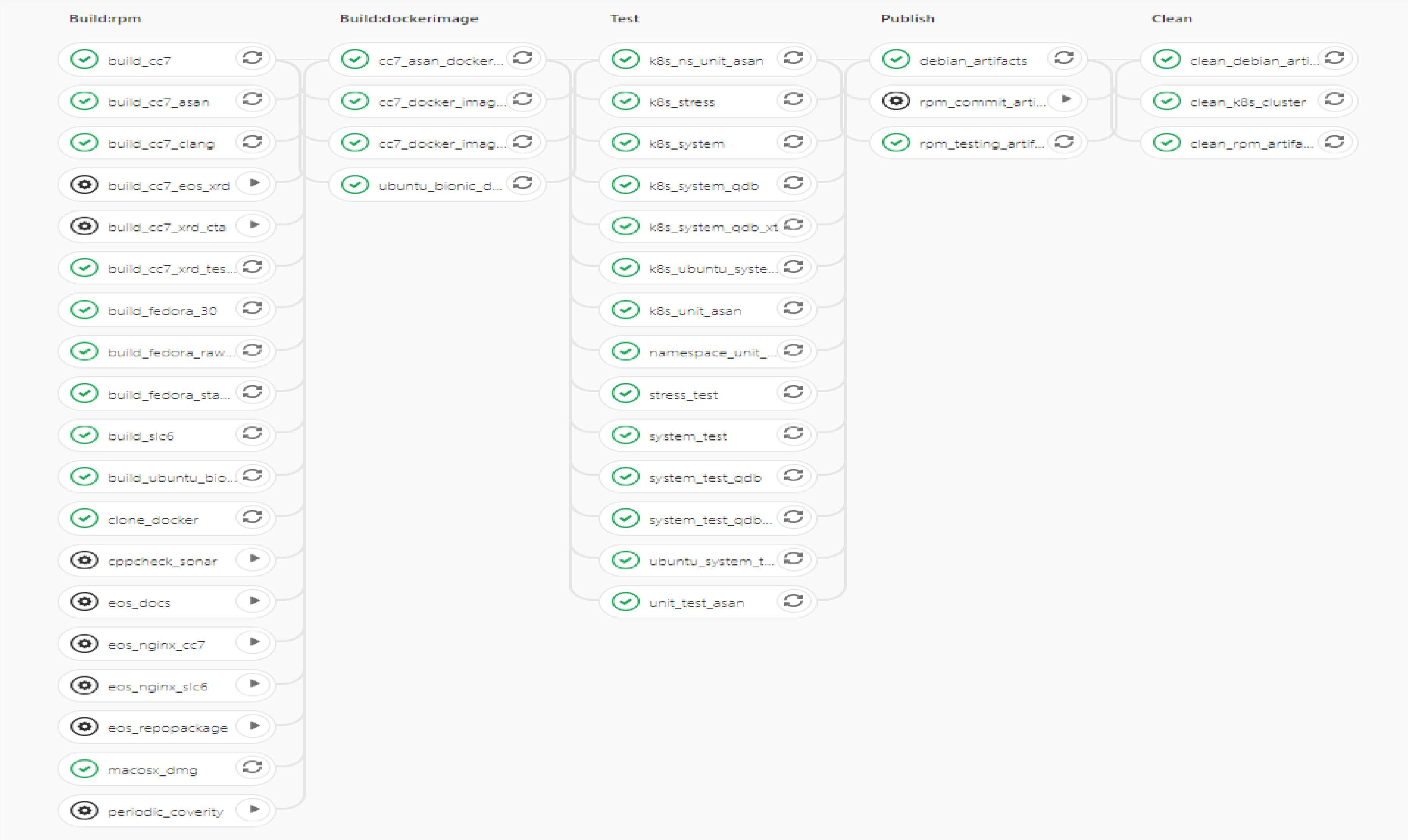
Goal: commit pipeline should be fast again

- Compile *non-production* builds just once per day
- Perform more intensive tests
- Implemented via templated `.gitlab-ci.yml`

… good place for *esoteric* builds

Nightly Builds – what changed?

Build		Docker		Testing	
cc7		cc7		system	
slc6		ubuntu_bionic		system_qdb	
macosx_dmg			cc7_xrd_testing	k8s_system	
ubuntu bionic			cc7_asan	k8s_system_qdb	
	cc7_xrd_testing			ubuntu_bionic	
	cc7_asan			stress	
	cc7_clang			k8s_stress	
	fedora_stable			unit_tests	
	fedora_30				unit_asan
	fedora_rawhide				ubuntu_disco
	ubuntu_disco				



Build:rpm

- build_cc7
- build_cc7_asan
- build_cc7_clang
- build_cc7_eos_xrd
- build_cc7_xrd_cta
- build_cc7_xrd_tes...
- build_fedora_30
- build_fedora_raw...
- build_fedora_sta...
- build_slc6
- build_ubuntu_bio...
- clone_docker
- cppcheck_sonar
- eos_docs
- eos_nginx_cc7
- eos_nginx_slc6
- eos_repopackage
- macosx_dmg
- periodic_coverity

Build:dockerimage

- cc7_asan_docker...
- cc7_docker_imag...
- cc7_docker_imag...
- ubuntu_bionic_d...

Test

- k8s_ns_unit_asan
- k8s_stress
- k8s_system
- k8s_system_qdb
- k8s_system_qdb_xt
- k8s_ubuntu_syste...
- k8s_unit_asan
- namespace_unit_...
- stress_test
- system_test
- system_test_qdb
- system_test_qdb...
- ubuntu_system_t...
- unit_test_asan

Publish

- debian_artifacts
- rpm_commit_arti...
- rpm_testing_artif...

Clean

- clean_debian_arti...
- clean_k8s_cluster
- clean_rpm_artifa...

Esoteric builds – asan

- Compile EOS with address sanitizer enabled
- Support provided via CMake and rpmbuild (tested only on CC7)
- Identified linking problems between shared and static libraries
- Run in CI unit tests → few problems discovered

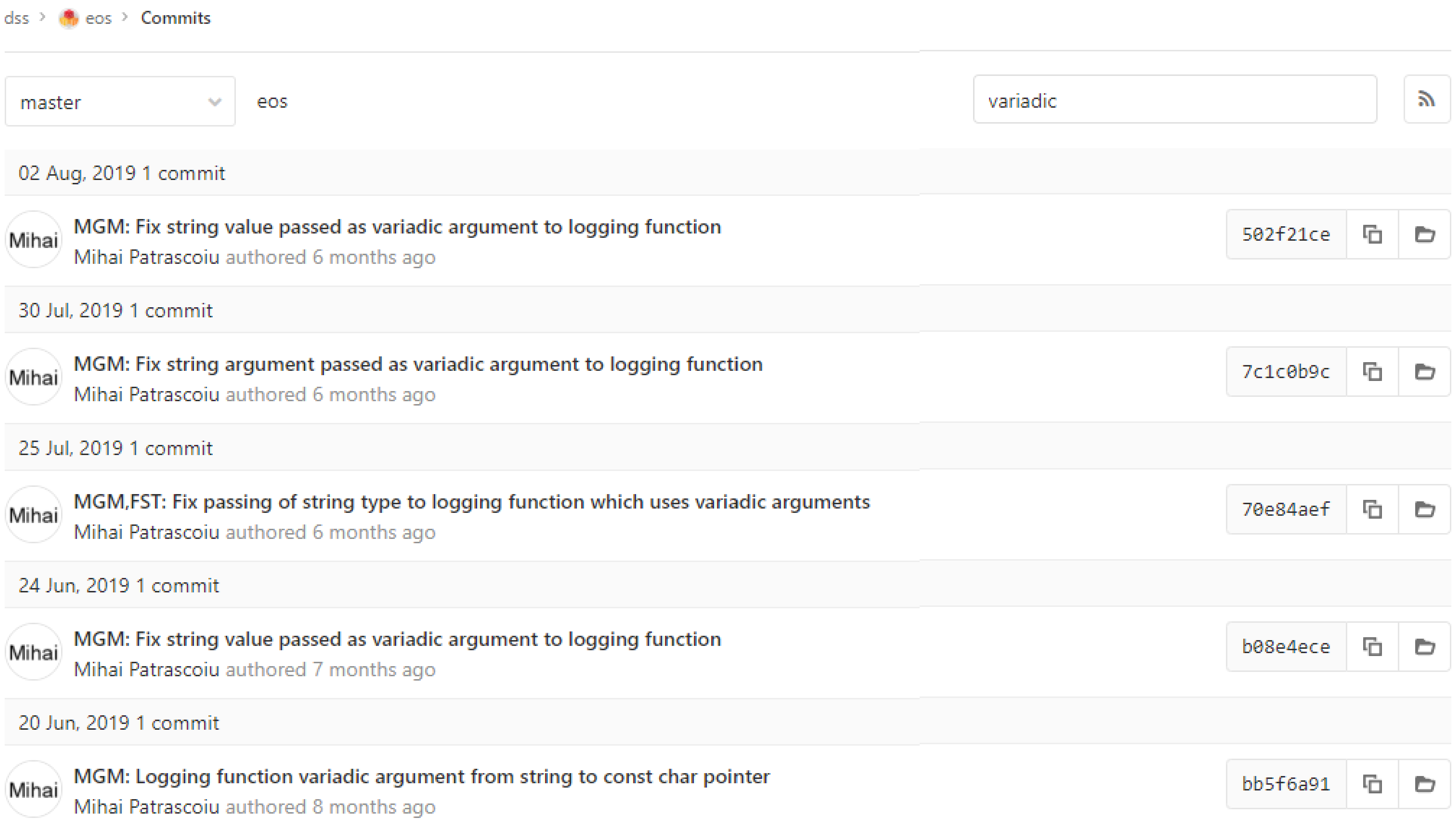
  (limited scope and good practices [collections, shared_pointers])

```
$ cmake3 ../ -DASAN=1     # <==> gcc -fsanitize=address

$ rpmbuild --with asan [..]
```

Esoteric builds – clang

- Replace gdb devtoolset with llvm-toolset
- Support provided via CMake and rpmbuild (tested only on CC7)

```
$ cmake3 ../ -DCLANG=1

$ rpmbuild --with clang [..]
```


dss › eos › Commits

master | eos | variadic

02 Aug, 2019 1 commit

Mihai MGM: Fix string value passed as variadic argument to logging function
Mihai Patrascoiu authored 6 months ago — 502f21ce

30 Jul, 2019 1 commit

Mihai MGM: Fix string argument passed as variadic argument to logging function
Mihai Patrascoiu authored 6 months ago — 7c1c0b9c

25 Jul, 2019 1 commit

Mihai MGM,FST: Fix passing of string type to logging function which uses variadic arguments
Mihai Patrascoiu authored 6 months ago — 70e84aef

24 Jun, 2019 1 commit

Mihai MGM: Fix string value passed as variadic argument to logging function
Mihai Patrascoiu authored 7 months ago — b08e4ece

20 Jun, 2019 1 commit

Mihai MGM: Logging function variadic argument from string to const char pointer
Mihai Patrascoiu authored 8 months ago — bb5f6a91

Conclusions

- Left unchecked, the CI entropy increases

  → constant effort to keep it in check
- Further improvements are possible (and desirable)

  → reduce build times even more by “upgraded-base” CC7 (Fabio Luchetti)

  → split testing into nightly
- Trying out different builds brings benefits

  → different compilers, different ~~errors~~ warnings

CI Updates

Code Coverage

Coverage build

- **Experimental** build that enables code coverage
- Uses gcov/lcov stack
- CMake and rpmbuild support

```
$ cmake3 ../ -DCOVERAGE=1 –DCOV_CROSS_PROFILE=1
$ make coverage-report

$ rpmbuild --with coverage [..]
```

Coverage `make` targets

```
make raw-code-trace            # lcov capture all coverage data
     filtered-trace-server # lcov capture only server
     filtered-trace-client # lcov capture only client

     coverage-server           # html report of server capture
     coverage-client           # html report of client capture
```

Note: certain subdirectories are filtered from the server capture
(console, unit_tests, 3rd party libraries)

Coverage mechanism in EOS

- Code coverage traces are printed at the end of binary execution

  (great for binaries, bad for EOS shared libraries)
- Note: can force flush by calling `__gcov_flush()`

- Implemented SIGPROF signal handlers in MGM, FST & NS libraries
  - Upon signal, call `__gcov_flush()`
  - Also call signal handler on all *coverage_plugin* libraries loaded by me

Coverage mechanism in EOS (cont'd)

- Coverage feature is only compiled/enabled in the coverage build
- Signal-handler must be enabled via `EOS_COVERAGE_REPORT` environment variable (may never be too safe)

```
# if unsure, don’t try on production server

$ kill –s SIGPROF $(pidof xrootd)
```

Coverage and the CI

- EOS is compiled using coverage option
- `-DCOV_CROSS_PROFILE` → defines coverage data & source directories

  → produces a separate `eos-coverage` RPM
- Coverage docker image is built & deployed in containers
- All tests are executed (system, fusex client, stress, unit)
- Coverage trace files are collected from each container
- Coverage traces files are aggregated into one using lcov
- Final HTML report is done

Coverage and the CI (cont’d)

- Exact process may be seen at:

  gitlab.cern.ch/eos/eos-docker/coverage/eos-coverage-ci.sh

- EOS CI coverage reports:

  storage-ci.web.cern.ch/storage-ci/eos-coverage-reports/

Thank you for your time!

CERN