Update on CI test suite deployment

LArSoft Coordination Meeting

December, 8th 2015

Vito Di Benedetto

Status of the tests on Jenkins system (I)

- The first test includes all **LArSoft** products plus **MicroBooNE** or **DUNE** products.
- The full chain of the Jenkins build using the **prof** qualifier takes less than **30 min**.

stage time(min)	checkout	build	make_test	install	ci_tests	TOTAL
MicroBooNE v04_30_00 with LArSoft						
debug	2.16	10.96	22.34	8.68	13.74	**60**
prof	1.80	13.05	4.16	4.05	3.19	**29**
DUNE35T v04_29_02 with LArSoft						
prof	1.93	9.55	3.73	1.87	0.53	**19**

- The **debug** build runs ~2 times slower than the **prof** build.
- The **make_test** and **ci_tests** stages have the bigger difference (4-5 times).
- MicroBooNE has 6 CI tests, DUNE has 5 CI tests; there is 1 test for each experiment production stage. These tests runs in parallel.
- N.B. Total time includes also dead time between stages.

Status of the tests on Jenkins system (II)

• This second test includes only **MicroBooNE** or **DUNE** products, the test uses prebuilt **LArSoft** products.

• The full chain of the Jenkins build using the **prof** qualifier takes **6-12 min**.

stage time(min)	checkout	build	make_test	install	ci_tests	TOTAL
MicroBooNE v04_30_01						
prof	0.66	2.42	4.23	0.35	2.96	**12**
DUNE35T v04_30_01						
prof	0.10	1.72	0.54	0.29	2.17	**6.2**

• MicroBooNE has 6 CI tests, DUNE has 5 CI tests; there is 1 test for each experiment production stage. These tests runs in parallel.

• N.B. Total time includes also dead time between stages.

Status of the tests on Jenkins system (III)

- This third test includes all **LArSoft** products plus **MicroBooNE** and **DUNE** products.
- The full chain of the Jenkins build using the **prof** qualifier takes **24 min**.

Build	Start Time	Platform	checkout	build	make_test	install	ci_tests
lar_ci_beta/1764	2015-12-08 02:49:28.515946	Linux 2.6.32-573.3.1.el6.x86_64					
	2015-12-08 02:53:20.908465	Darwin 13.4.0					
lar_ci_beta/1763	2015-12-08 01:18:26.446004	Linux 2.6.32-573.3.1.el6.x86_64					
	2015-12-08 01:21:18.506338	Darwin 13.4.0					

Progress Legend: Running, Pending, Succeeded, Failed, Skipped

Stage	checkout	build	make_test	install	ci_tests	TOTAL
MicroBooNE + DUNE35T + LarSoft v04_30_02 [execution time(min)]						
prof	2.65	9.54	3.80	1.82	4.63	**24**

- In the ci_tests stage there are 6 MicroBooNE CI tests plus 5 DUNE CI tests.
- These tests are independent and they runs in parallel.
- In the previous test the ci_test stage runs in about 3 min, likely the time increase is due to changes in the code (?).
- Adding CI regression tests for other experiment is not supposed to increase the running time; CI regression tests are independent and they can run in parallel.
- N.B. Total time includes also dead time between stages.
- **This configuration is supposed to be used in production.**

CI test suite development status

- Improved CI test configuration ci_tests.cfg:
  - now experiment ci_tests.cfg's don't interfere each other;
  - CI tests for different experiments can run together;
- CI regression test implementation available in the experiments *feature/vdb_ci_regression_test_suite* feature branch.
  - The feature branch is updated to the today experiment develop branch for MicroBooNE and DUNE.
- In the lar_ci repository added/updated some configuration files to run the CI regression test suite for MicroBooNE and DUNE in the same Jenkins build.
  - This work is included in the *vitoBranch* of the lar_ci repository.

CI test suite TODO list

- Merge the feature branch *feature/vdb_ci_regression_test_suite* into the develop branch of the experiment,
  - for MicroBooNE and DUNE it is ready.

- Merge the *vitoBranch* into lar_ci master branch (?),
  - eventually cut a tag and use it as default lar_ci tag for the Jenkins build.

- Update the post-receive git hook to **trigger** the CI regression test suite **quick_test** instead of the actual **default** CI test suite.
  - Permission required to log into the repository machine (?)

Backup slides

LArSoft CI regression test suite

- **CI regression test suite implemented for:**
  ArgoNeuT, DUNE35T, LArIAT, MicroBooNE, SBDN.
  - The code is published in the feature branch *feature/vdb_ci_regression_test_suite* for each experiment repository.
- **Implementation details:**
  - The CI regression test runs all the stages of the experiment work-flow.
  - Runs updated/modified code to generate “current” data files.
  - Use official FHiCL files with some option added to handle the random seed for the CI test.
  - Compare “current” data files against “reference” data files generated for this purpose.
    - Reference files are located in the experiment scratch dCache area.

CI test implementation details

Test example:

srcs/uboonecode/test/ci/ci_tests.cfg excerpt

```
[DEFAULT]
STEPS = none gen geant detsim reco1 reco2 ana
LARSOFT_REFERENCE_VERSION=v04_20_00
BASEFILENAME=prodgenie_bnb_nu_cosmic_uboone
EXPCODE=uboonecode
EXPSCRIPT=ci_regression_test_uboonecode.sh
INPUTFILEDIR=/pnfs/uboone/scratch/users/vito/ci_tests_inputfiles
```

Experiment specific configuration

```
[test ci_gen_regression_test_uboonecode]
script=${UBOONECODE_DIR}/test/%(EXPSCRIPT)s
STEP=1
NEVENTS=1
args=%(NEVENTS)s %(STEP)s %(LARSOFT_REFERENCE_VERSION)s %(BASEFILENAME)s %(EXPCODE)s %(STEPS)s
inputfiles=%(INPUTFILEDIR)s/%(BASEFILENAME)s_Reference_gen_%(LARSOFT_REFERENCE_VERSION)s.root %
(INPUTFILEDIR)s/GenRandomSeeds_Ref.dat
parse_art_output=True
mem_usage_range=100:20000000
cpu_usage_range=10:60000
...
```

CI test section

```
[suite quick_test_uboonecode]
testlist=ci_gen_regression_test_uboonecode ci_geant_regression_test_uboonecode
ci_detsim_regression_test_uboonecode ci_reco1_regression_test_uboonecode ci_reco2_regression_test_uboonecode
ci_ana_regression_test_uboonecode
```

CI test suite section

- The script to run the test is the same for all experiments.
- The “experiment specific section” in the ci_tests.cfg sets all required input to properly initialize the script.
- The “CI test section” sets further arguments for the specific CI test.
- The “CI test suite section” collects a list of tests to run all together.

CI regression test statistics

Experiment	qualifier	stage	Time (min)	Memory (Mb)
MicroBooNE	prof v04_14_00	prodgenie	2:04	591
		g4	2:36	1665
		detsim	0:59	1190
		reco1	1:13	1338
		reco2	1:55	1410
		analysis	0:47	1410
	prof v04_22_00	prodgenie	2:15	609
		g4	2:53	1840
		detsim	1:17	1392
		reco1	1:35	1464
		reco2	2:46	1506
		analysis	0:56	1508

Tests executed on uboonegpvm03

- Simulated a ~1.8GeV ν_μ interaction with cosmic rays.
- MicroBooNE code uses similar resources in both LArSoft releases.
- Each test runs well within the target time upper limit (10 min)

CI regression test statistics

Experiment	qualifier	stage	Time (min)	Memory (Mb)
DUNE35T	prof v04_14_00	prodantimu	0:09	89
		g4	0:18	336
		detsim	0:13	164
		reco	0:15	252
		analysis	0:27	4543
	prof v04_22_00	prodantimu	0:09	89
		g4	0:13	215
		detsim	0:13	152
		reco	0:27	329
		analysis	0:14	7664

Tests executed on lbnegpvm03

- Simulated a ~1.6 GeV single $\bar{\mu}$ interaction.
- DUNE35T code uses similar resources in both LArSoft releases.
- Each test runs well within the target time upper limit (10 min)

CI regression test statistics

Experiment	qualifier	stage	Time (min)	Memory (Mb)
LArIAT	prof v04_14_00	fragment	0:11	204
		reco	0:16	211
	prof v04_22_00	slicer	0:14	165
		beamlinereco	0:09	101
		reco2D	0:12	134

Tests executed on lariatgpvm03

- Procesed 1 real data event taken at FTBF, MCenter beamline, 16GeV beam.
- LArIAT changed FhiCL files between v04_14_00 and v04_22_00.
- Each test runs well within the target time upper limit (10 min).

CI regression test statistics

ArgoNeuT and SBND FHiCL files used for this CI test are splitted to allow independ checks for each simulation step

Experiment	qualifier	stage	Time (min)	Memory (Mb)
ArgoNeuT	prof v04_14_00	sim	1:33	682
		reco	0:08	190
	prof v04_22_00	gen	1:29	636
		geant	0:13	140
		detsim	0:09	101
		reco	0:10	192

Tests executed on argoneutgpvm03

- Simulated a ~2GeV ν_μ interaction.
- ArgoNeuT uses similar resources in both LarSoft releases.
- All tests run well within the target time upper limit (10 min).

Experiment	qualifier	stage	Time (min)	Memory (Mb)
SBND	prof v03_08_02	sim	2:07	540
		reco	0:35	713
	prof v04_00_00	gen	0:13	87
		geant	2:06	364
		detsim	0:14	278
		reco	Not Available	Not Available

Tests executed on lar1ndgpvm01

- Simulated a ~2GeV single **μ** interaction.
- SBND uses similar resources in both LarSoft releases.
- All tests run well within the target time upper limit (10 min).
- reco stage can't run because of some issue, reco statisctic not yet available.