

# FluDAG Continuous Integration

Overview

# FluDAG Continuous Integration

- Nightly vs Pull-Request
- 'develop' branch vs feature branch
- Manual launch vs 'polyphemer' automatic launch
- build only vs build-and-test ('test')
- BaTLab 'dagmcci' account vs personal BaTLab account
- Github: ssh vs https
- Environment:
  - Use of submit.sh to build .scp files on the fly
- Submit node vs. Run node vs. Execute node
- Platform complications: gcc versions, cmake versions, order of compile

# FluDAG Continuous Integration

Purpose

# FluDAG CI

## Purpose

- verify code builds on different platforms
- run unit tests
- view unit test results
- protect 'develop' branch from build-breaking merges
- verify 3<sup>rd</sup> party code updates don't break

# FluDAG CI

## BaTLab

- Allow user to write and run build scripts from their account
- Use shell scripts, formatted instruction scripts, and 'Metronome' commands: *nmi\_xxx*
- Tutorial at [www.batlab.org](http://www.batlab.org)
- Go to [www.batlab.org](http://www.batlab.org) to create an account, e.g. `jczachman@submit-1.batlab.org`

# FluDAG CI more BaTLab

- Run-spec file can have lines to cause notifications to specific email addresses.
- Email notification can be set to 'notify-on-fail'
- BaTLab node doesn't talk nicely with github https url, must clone repo with ssh and add sshkeys of login account to github repo.

# FluDAG Continuous Integration

Nightly Build

# FluDAG CI

## Nightly Build I

- Run-specification file sets time of day to run
  - This is the only substantive difference with the PR run-spec file
- Normally runs are submitted with Metronome command,  
`>nmi_submit fludag_nightly.run-spec`
- [dagmcci@submit-1.batlab.com](mailto:dagmcci@submit-1.batlab.com) is the launch account



# FluDAG CI

## Nightly Build II

- FluDAG build scripts reside in [github.com/svalinn/DAGMC-CI](https://github.com/svalinn/DAGMC-CI)
- Configurable daily email contains url of build results, e.g.  
<http://submit-1.batlabs.org/nmi/results/details?runID=236185>
- The nightly build has to be removed if it needs to be updated

# fludag\_nightly.run-spec, p. 1

```
run_type = test
```

```
inputs = fetch/hdf5.url, fetch/moab.url,  
fetch/dagmc.git, fluka.scp, run-test.scp,  
generate_test_list.scp, gcc.SL6.scp,  
build.scp, build.SL6.scp
```

```
x86_64_Debian7_remote_pre_declare = build.sh
```

```
x86_64_Ubuntu12_remote_pre_declare = build.sh
```

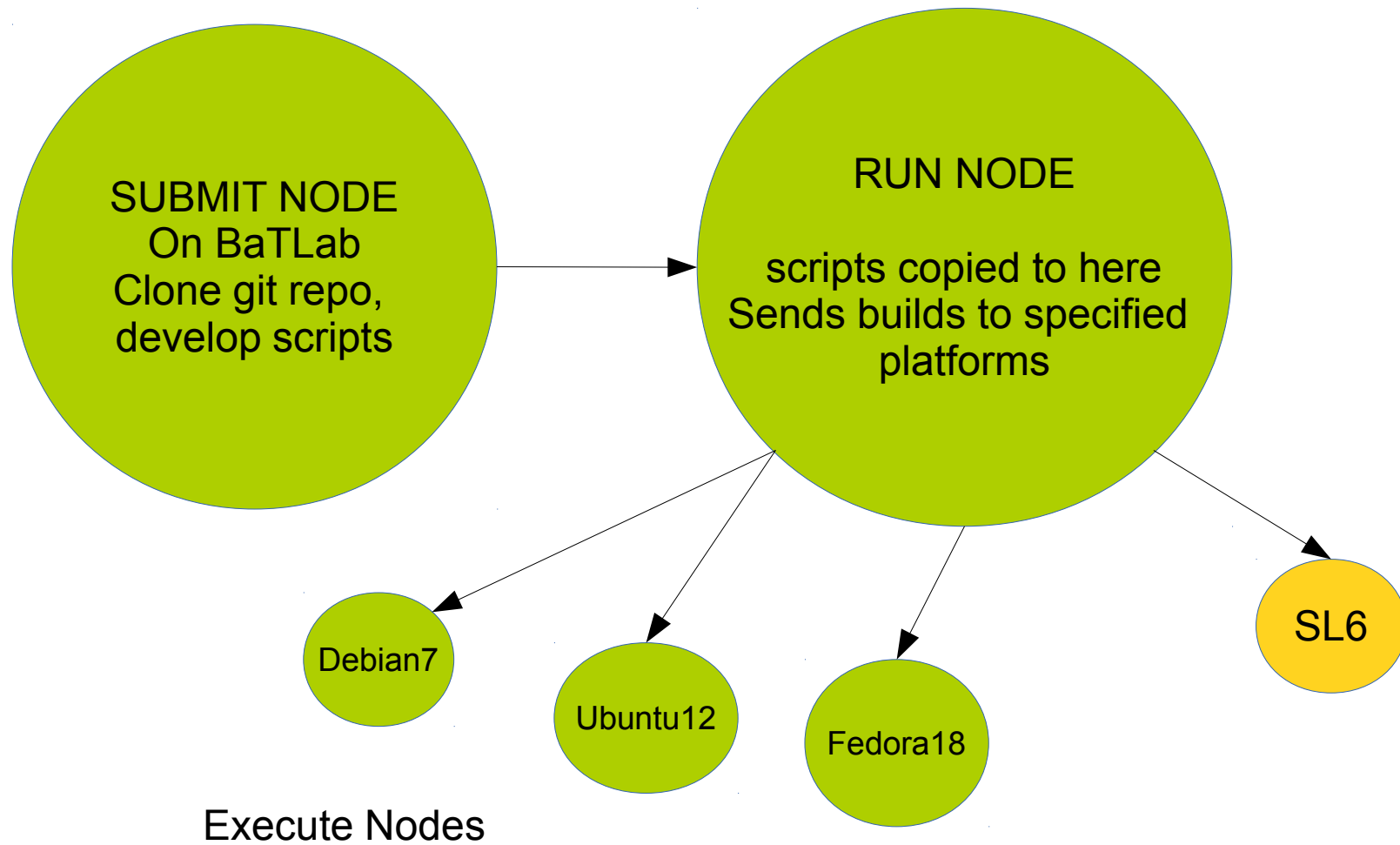
```
x86_64_Fedora18_remote_pre_declare = build.sh
```

```
remote_declare = generate_test_list.sh
```

# fludag\_nightly.run-spec, p. 2

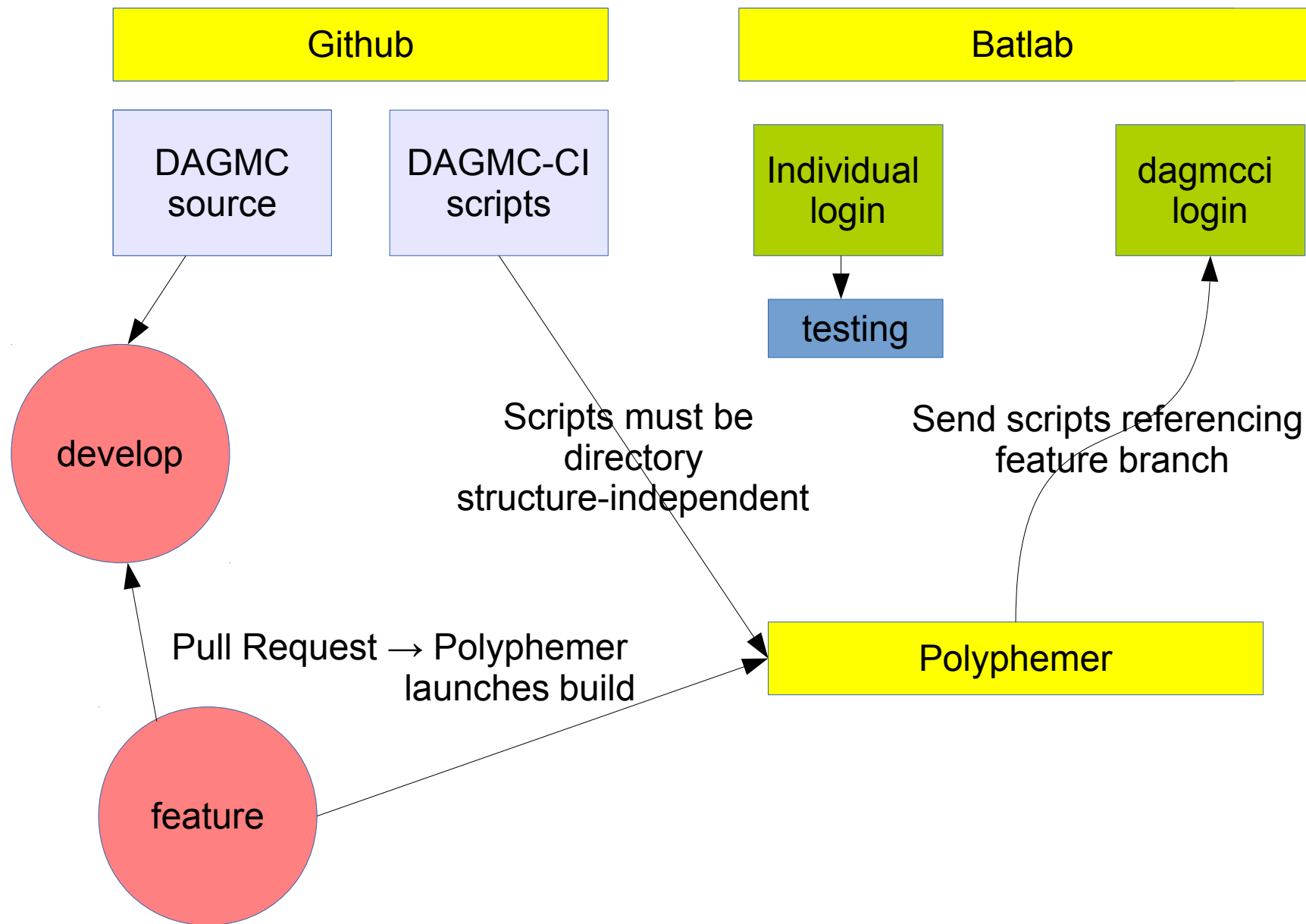
```
remote_task      = run-test.sh
platforms        = x86_64_Debian7,
x86_64_Ubuntu12, x86_64_Fedora18
project          = Fludag Nightly Unit Testing
description      = Run tests on all known good
platforms
notify           = dagmcci@googlegroups.com
cron_hour        = 14
cron_minute      = 15
```

## BaTLab Structure



# FluDAG Continuous Integration

Pull Request Build



# FluDAG CI Pull Request I

Polyphemer server causes a build on the feature branch when a pull request to 'develop' is created

- BaTLab scripts are automatically pulled from svalinn DAGMC-CI master branch
- Scripts are placed in a subdirectory of BaTLab dagmcci account with a name assigned as “svalinn—DAGMC—*pullRequest#*”

```
[dagmcci@submit-1 ~]$ ls
```

```
fluka          svalinn--DAGMC--200  svalinn--DAGMC--204  svalinn--DAGMC--208  svalinn--DAGMC--212  svalinn--DAGMC--216
gcc_SL5        svalinn--DAGMC--201  svalinn--DAGMC--205  svalinn--DAGMC--209  svalinn--DAGMC--213  svalinn--DAGMC--219
gcc_SL6        svalinn--DAGMC--202  svalinn--DAGMC--206  svalinn--DAGMC--210  svalinn--DAGMC--214  tars
repository     svalinn--DAGMC--203  svalinn--DAGMC--207  svalinn--DAGMC--211  svalinn--DAGMC--215
```

# FluDAG CI

## Pull Request II

- Bash scripts that must refer to the local directory structure under \$HOME will be incorrect
- Script submit.sh creates these files on the fly and then calls nmi\_submit. Note that Polyphemer must be told to run submit.sh
- BaTLab disk full ==> FAIL



# FluDAG CI

## Pull Request III

- Polyphemer resides at [github.com/polyphemer-ci/polyphemer](https://github.com/polyphemer-ci/polyphemer)
- `batlabrun.py` is the main script
- In the input specification file that clones the DAGMC repo and checks out the branch to build, polyphemer runs '`sed -i ...`' to substitute the PR branch

# FluDAG Continuous Integration

Platforms

# FluDAG CI Platforms

- Nightly and PR builds run on Fedora18, Ubuntu12 and Debian7
- Scientific Linux 6 gcc is too old for proper fluka library linking
  - Newer version of gcc has been built with a different set of scripts and made available to copy from submit node
  - SL6 also has older version of cmake => cmake scripts must not name a later version

# FluDAG CI

## Did I cover everything?

- Nightly vs Pull-Request
- 'develop' branch vs feature branch
- Manual launch vs 'polyphemer' automatic launch
- build only vs build-and-test ('test')
- BaTLab 'dagmcci' account vs personal BaTLab account
- Github: ssh vs https
- Environment:
  - Use of submit.sh to build .scp files on the fly
- Submit node vs. Run node vs. Execute node
- Platform complications: gcc versions, cmake versions, order of compile