

## 2018 International Radiance Workshop

# Automatic for the People - Automated Build/Test/Packaging for Radiance

Rob Guglielmetti

[National Renewable Energy Laboratory - Golden, CO USA]

# Outline

## Continuous Integration (CI) for Radiance

- Motivation, definition, etc.
- Implementation
  - Workflow
  - GitHub Mirror
  - GitLabCI
- Case Study
- Next Steps

# Motivation

# Motivation

- OpenStudio project (NREL 2010)
- Cross platform, most users on Windows
- Other efforts:
  - Axel Jacobs (last update ~2016)
  - Francesco Anselmo (mingw, ~2008, no rvu, etc)
  - Georg Mischler (Rayfront ~2002, Windows rvu!)
  - LBNL Desktop Radiance (2000?)

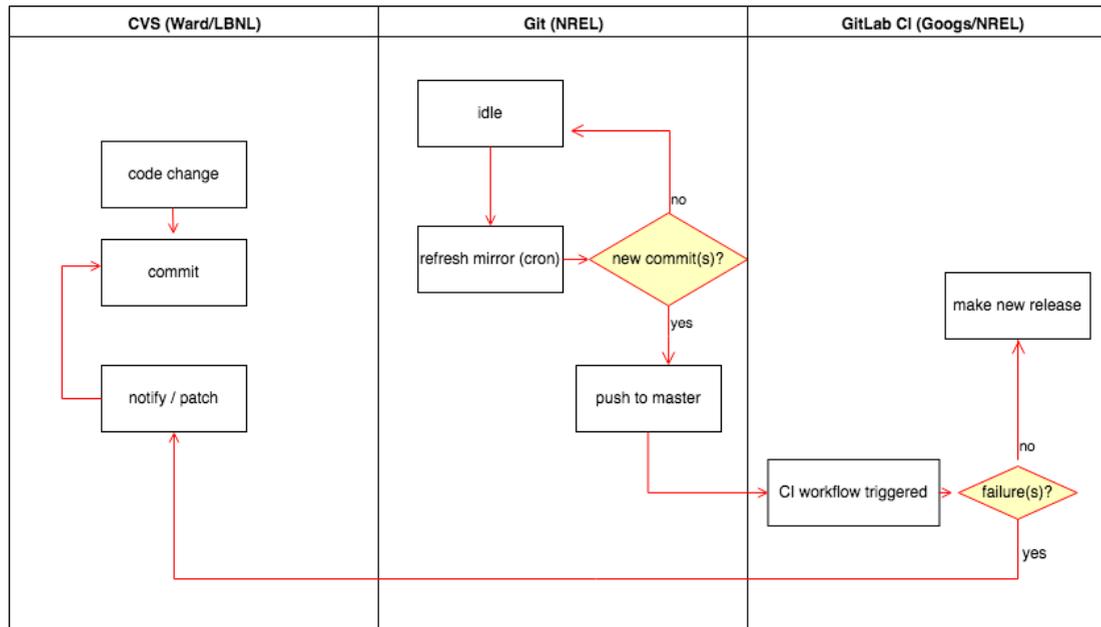
**Needed easy way to produce reliable, cross-platform Radiance installers,  
keep up with HEAD**

- Windows versions of GUI apps e.g. rvu
- Releases tied to OpenStudio releases
- CMake build system (implemented ~2013)

# How it Works

[GitHub, GitlabCI, CMake]

# Workflow

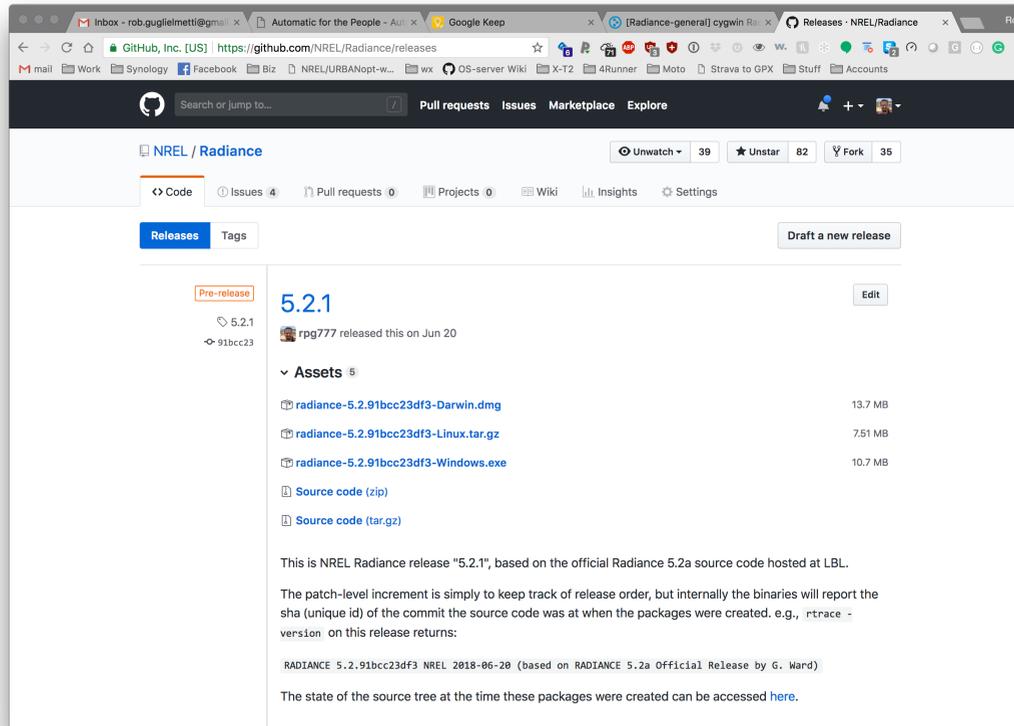


# GitHub Mirror of CVS

The screenshot shows the GitHub repository page for NREL/Radiance. The repository is a mirror of the master Radiance cvs source repo, used for the creation of Radiance installers for the NREL OpenStudio project. It has 6,050 commits, 3 branches, 27 releases, and 2 contributors. The latest commit is by Gregory J. Ward, adding a missing file from the last commit, made a day ago. The commit history includes updates to CMake build systems, SCons options, and package installers.

Commit	Message	Time
Gregory J. Ward	Added missing file from last commit	Latest commit a46558b a day ago
InstallRules	Update to CMake build system from Rob G. and Alex S., adding test cases	8 months ago
build_utils	SCons Option to build Pmap with OOC data on unix	8 months ago
doc	Added missing file from last commit	a day ago
platform	Updating to SCons 3.x, adding support for Python 3	8 months ago
resources	Fixes to changes that were not applied fully the first time for CMake	8 months ago
src	Added missing file from last commit	a day ago
test	Fixes to changes that were not applied fully the first time for CMake	8 months ago
.gitignore	Added missed files from CMake update	8 months ago
.gitlab-ci.yml	Added missed files from CMake update	8 months ago
CMakeLists.txt	Updates to package installers from Rob G.	2 months ago
INSTALL.scons	SCons Option to build Pmap with OOC data on unix	8 months ago
License.txt	Changes for 5.0 release	3 years ago
MANIFEST	Added files needed by cmake	7 years ago

# GitHub Releases Page



The screenshot shows the GitHub Releases page for the repository NREL/Radiance. The page is viewed in a browser with several tabs open, including 'Automatic for the People - Aut...', 'Google Keep', '[Radiance-general] cygwin Re...', and 'Releases - NREL/Radiance'. The URL in the address bar is <https://github.com/NREL/Radiance/releases>. The repository name 'NREL / Radiance' is displayed at the top, along with statistics: 39 Unwatch, 82 Unstar, and 35 Fork. The 'Releases' tab is selected, and a 'Draft a new release' button is visible. The current release is 5.2.1, marked as a 'Pre-release', released on Jun 20 by user rpg777. It includes three assets: 'radiance-5.2.91bcc23df3-Darwin.dmg' (13.7 MB), 'radiance-5.2.91bcc23df3-Linux.tar.gz' (7.51 MB), and 'radiance-5.2.91bcc23df3-Windows.exe' (10.7 MB). Source code is available as 'Source code (zip)' and 'Source code (tar.gz)'. The release description states: 'This is NREL Radiance release "5.2.1", based on the official Radiance 5.2a source code hosted at LBL. The patch-level increment is simply to keep track of release order, but internally the binaries will report the sha (unique id) of the commit the source code was at when the packages were created. e.g., `rtrace -version` on this release returns: 

```
RADIANCE 5.2.91bcc23df3 NREL 2018-06-20 (based on RADIANCE 5.2a Official Release by G. Ward)
```

 The state of the source tree at the time these packages were created can be accessed [here](#).'

# GitHub Issues Page

The screenshot shows the GitHub Issues page for the NREL/Radiance repository. The browser address bar displays the URL <https://github.com/NREL/Radiance/issues>. The repository name "NREL / Radiance" is visible at the top, along with statistics: 39 Unwatch, 82 Unstar, and 35 Fork. The "Issues" tab is selected, showing 4 issues. A filter is applied: "is:issue is:open". The issues list includes:

- 4 Open** / 14 Closed
- Test Perl version of optics2rad, add to packages** (opened Apr 13, 2017 by rpg777)
- Compiling 5.1 on linux** (opened Aug 31, 2017 by rgsouthall, 6 comments)
- Some .cal files missing from release** (opened Oct 19, 2016 by markstock, 1 comment)
- Radiance package ununinstaller clobbers RAYPATH** (opened May 11, 2015 by rpg777, labeled as a bug)

A ProTip at the bottom indicates: "Updated in the last three days: updated:>2018-08-30."

## GitHub Issues Page (cont'd.)

The screenshot shows a GitHub issue page for the repository 'NREL / Radiance'. The issue title is 'Some .cal files missing from release #15'. The issue is open and was created by 'markstock' on Oct 19, 2016. There is one comment from 'rpg777' on Apr 2, 2016, which is self-assigned. The issue has 39 watchers, 82 unstars, and 35 forks. The right sidebar shows the issue is assigned to 'rpg777', has no labels, projects, milestones, or notifications. There are 2 participants in the discussion.

Some .cal files missing from release #15

markstock opened this issue on Oct 19, 2016 · 1 comment

markstock commented on Oct 19, 2016

I was running my benchmark on a new machine after installing the 5.0.a.12 binary release, and it seems that picture.cal and dirt.cal are missing from radiance/lib. I didn't do an exhaustive search for other missing files.

rpg777 self-assigned this on Apr 2

rpg777 commented on Apr 2

Indeed @markstock, those files are not in the source tree, nor are they in the "supplemental files archive" anymore. So needless to say they are not getting baked into the NREL packages. Not sure where they went or when. Perhaps we could take this over to radiance-dev?

Write Preview

Leave a comment

Assignees: rpg777

Labels: None yet

Projects: None yet

Milestone: No milestone

Notifications: Unsubscribe

You're receiving notifications because you were assigned.

2 participants

Lock conversation

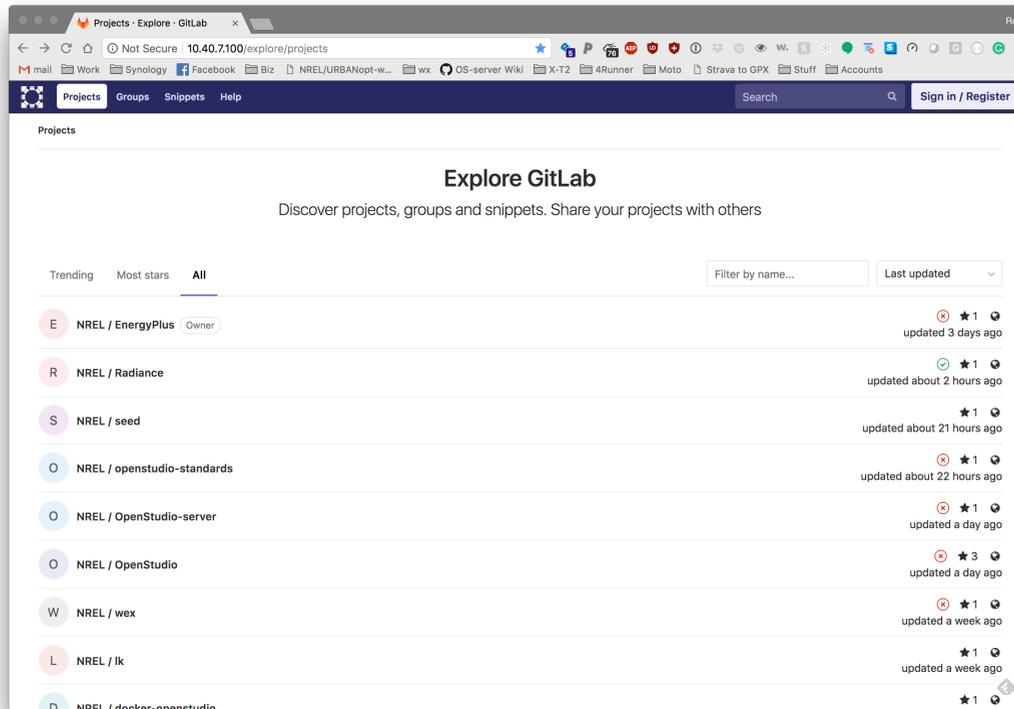
# GitLabCI

## Continuous Integration

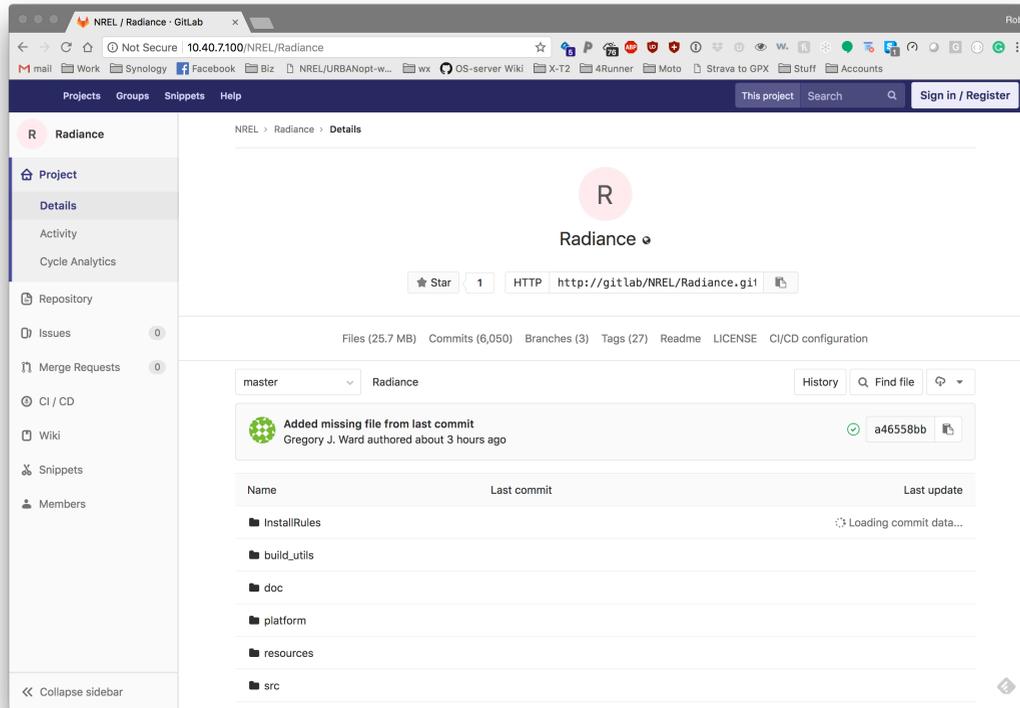
- Monitors Git Repo
- Any change triggers new build/test/deploy



# GitLabCI - Projects Page (overview)



# GitLabCI - Project Detail



# The GitLabCI Experience

[and a case study]

## Prologue

**From:** Jan  
**To:** Greg, Rob  
**Date:** Friday, Aug 31 2018

Hi Greg,

I have some small changes to evalglare (change of default values) **and** an extended version of gendaylit (color option included). Both should compile without problems (I tested it **and** had also only small changes).

If you have time, please **include** them into the CVS.

**Important:** gendaylit needs now a second cal file (**for** the colored option, so please add the perezlum\_c.cal to the lib folder).

Both tools have also updated manpages which are also included **in** the tar files.

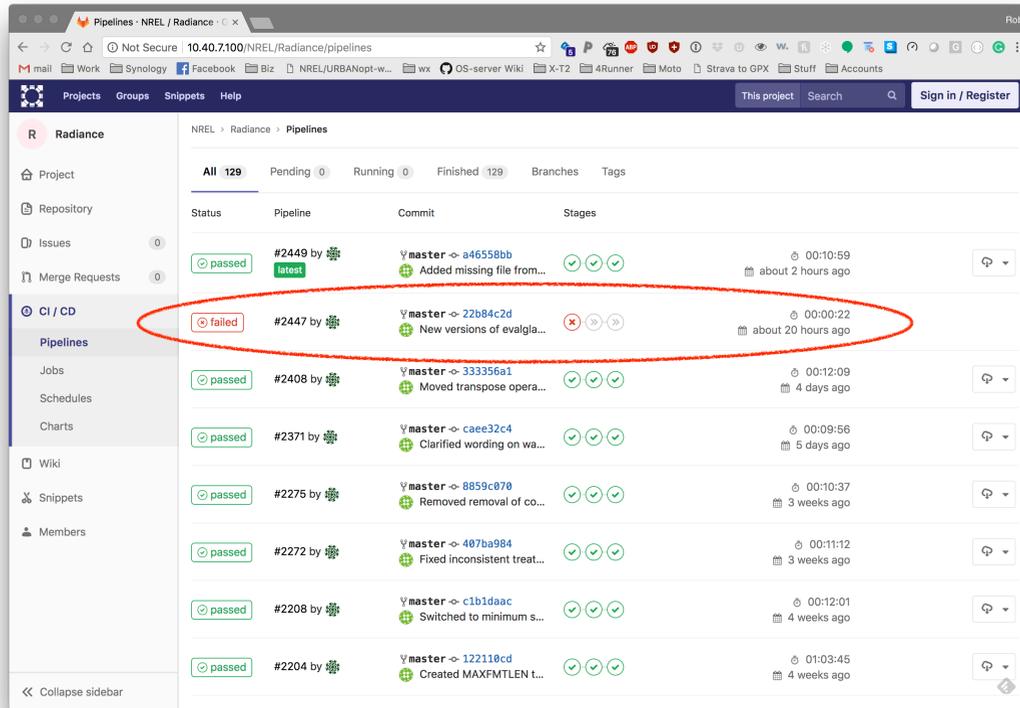
Thx!

See you on Monday!

best

Jan

# GitLabCI - Build fail



# GitLabCI - Build fail - STDOUT

The screenshot shows a GitLab CI build job for the 'Radiance' project on a Windows environment. The build process is failing due to a missing TIFF library. The terminal output is as follows:

```

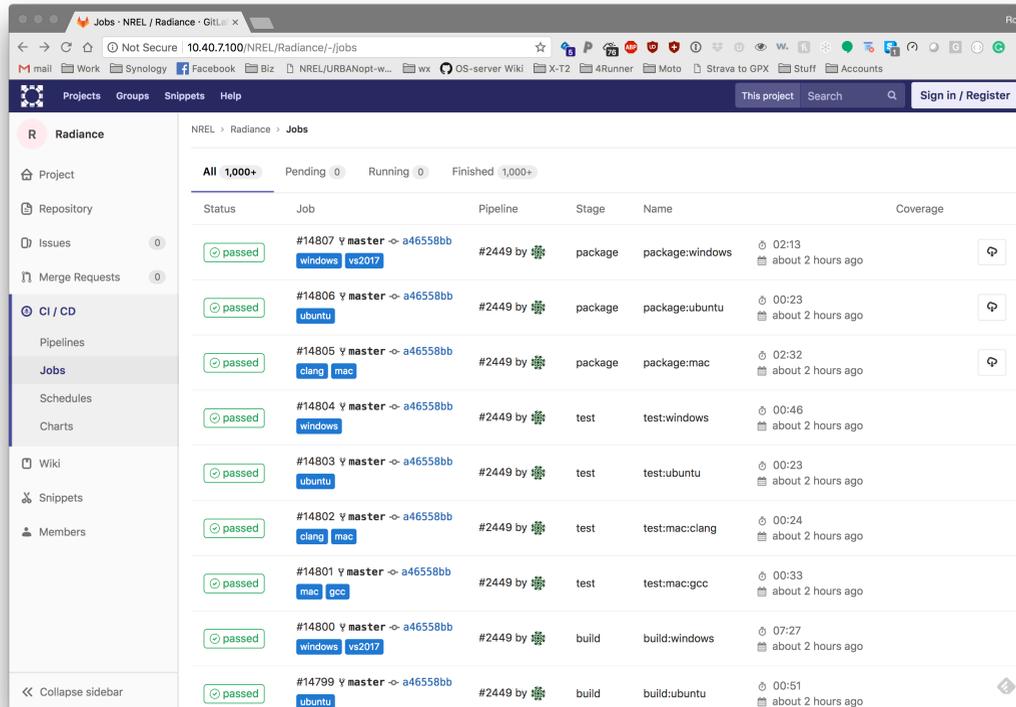
Using Shell executor...
Running on ASWINDLE-W10VM...
Fetching changes...
Removing build/
HEAD is now at 333356a11 Moved transpose operation later to save memory with -c (transform) reduction
From http://gitlab/NREL/Radiance
  333356a11..22b84c2d4  master    -> origin/master
Checking out 22b84c2d4 as master...
Skipping git submodule setup
$ set PATH=APATH;C:\Qt5.10.0\msvc2017_64\bin
$ mkdir build
$ cd build
$ "C:\Program Files\Make\bin\cmake.exe" -G "Visual Studio 15 2017 Win64" -DBUILD_TIFF=ON ..
-- The C compiler identification is MSVC 19.12.25334.0
-- The CXX compiler identification is MSVC 19.12.25834.0
-- Check for working C compiler: C:/Program Files (x86)/Microsoft Visual Studio/2017/Enterprise/VC/Tools/MSVC/14.12.25827/bin/Hostx86/x64/cl.exe
-- Check for working C compiler: C:/Program Files (x86)/Microsoft Visual Studio/2017/Enterprise/VC/Tools/MSVC/14.12.25827/bin/Hostx86/x64/cl.exe -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working CXX compiler: C:/Program Files (x86)/Microsoft Visual Studio/2017/Enterprise/VC/Tools/MSVC/14.12.25827/bin/Hostx86/x64/cl.exe
-- Check for working CXX compiler: C:/Program Files (x86)/Microsoft Visual Studio/2017/Enterprise/VC/Tools/MSVC/14.12.25827/bin/Hostx86/x64/cl.exe -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found Git: C:/Program Files/Git/cmd/git.exe (found version "2.15.1.windows.2")
Make Error at src/gen/MakeLists.txt:74 (file):
  file COPY cannot find
  "C:/GitLab-Runner/builds/c1016289/0/NREL/Radiance/src/gen/perezlum_c.ca!".

-- Could NOT find TIFF (missing: TIFF_LIBRARY TIFF_INCLUDE_DIR)
-- Found OpenGL: opengl32
-- Found Perl: C:/Program Files/Git/usr/bin/perl.exe (found version "5.24.1")
-- Configuring incomplete, errors occurred!
See also "C:/GitLab-Runner/builds/c1016289/0/NREL/Radiance/build/MakeFiles/MakeOutput.log".
ERROR: Job failed: exit status 1

```

The error message is circled in red in the original image. The build job is named 'build' and is part of a pipeline named 'build:windows'.

# GitLabCI - CVS add



The screenshot displays the GitLab CI/CD interface for the 'Radiance' project. The left sidebar shows the project structure, with 'CI / CD' selected. The main content area shows a list of jobs, all of which are in a 'passed' status. The jobs are organized by pipeline and stage.

Status	Job	Pipeline	Stage	Name	Coverage
passed	#14807 ✓ master → a46558bb windows vs2017	#2449 by [user]	package	package:windows	02:13 about 2 hours ago
passed	#14806 ✓ master → a46558bb ubuntu	#2449 by [user]	package	package:ubuntu	00:23 about 2 hours ago
passed	#14805 ✓ master → a46558bb clang mac	#2449 by [user]	package	package:mac	02:32 about 2 hours ago
passed	#14804 ✓ master → a46558bb windows	#2449 by [user]	test	test:windows	00:46 about 2 hours ago
passed	#14803 ✓ master → a46558bb ubuntu	#2449 by [user]	test	test:ubuntu	00:23 about 2 hours ago
passed	#14802 ✓ master → a46558bb clang mac	#2449 by [user]	test	test:mac:clang	00:24 about 2 hours ago
passed	#14801 ✓ master → a46558bb mac gcc	#2449 by [user]	test	test:mac:gcc	00:33 about 2 hours ago
passed	#14800 ✓ master → a46558bb windows vs2017	#2449 by [user]	build	build:windows	07:27 about 2 hours ago
passed	#14799 ✓ master → a46558bb ubuntu	#2449 by [user]	build	build:ubuntu	00:51 about 2 hours ago

# GitLabCI - Build Success

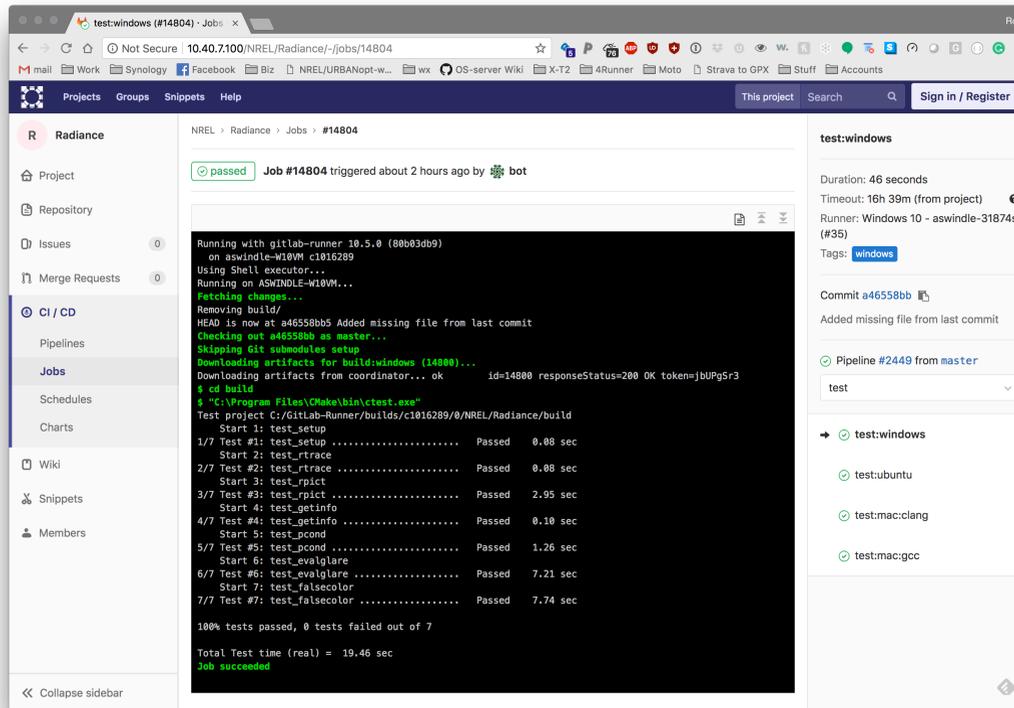
The screenshot displays the GitLab CI interface for a job named 'build:macclang' (ID #14798). The job is shown as successful. The left sidebar contains navigation options for the 'Radiance' project, including Project, Repository, Issues, Merge Requests, CI/CD (selected), Pipelines, Jobs, Schedules, Charts, Wiki, Snippets, and Members. The main area shows the job's progress and logs, which include the following output:

```

Scanning dependencies of target tiff-palette
[ 90%] Building C object test/MakeFiles/ascii_tag.dir/ascii_tag.c.o
[ 91%] Linking C executable strip_rw
[ 91%] Built target tiff-rgb
[ 92%] Building C object contrib/dbs/MakeFiles/tiff-bi.dir/tiff-bi.c.o
[ 93%] Building C object contrib/dbs/MakeFiles/tiff-palette.dir/tiff-palette.c.o
[ 93%] Building C object contrib/addtiff/MakeFiles/addtiff.dir/addtiff.c.o
Scanning dependencies of target tiff-grayscale
[ 94%] Linking C executable ascii_tag
[ 95%] Linking C executable tiff-bi
[ 96%] Linking C executable tiff-palette
[ 96%] Built target strip_rw
Scanning dependencies of target iptcutil
[ 97%] Linking C executable addtiff
[ 97%] Built target ascii_tag
[ 98%] Building C object contrib/iptcutil/MakeFiles/iptcutil.dir/iptcutil.c.o
[ 98%] Built target tiff-palette
[ 98%] Built target addtiff
[ 99%] Building C object contrib/dbs/MakeFiles/tiff-grayscale.dir/tiff-grayscale.c.o
[ 99%] Linking C executable iptcutil
[ 99%] Built target tiff-bi
[ 99%] Linking C executable tiff-grayscale
[ 99%] Built target iptcutil
[ 99%] Built target tiff-grayscale
[100%] Linking C executable tiffcrop
[100%] Built target tiffcrop
[100%] No install step for 'libtiff'
[100%] Completed 'libtiff'
[100%] Built target libtiff
Scanning dependencies of target ra_tiff
Scanning dependencies of target normtiff
[100%] Building C object src/px/MakeFiles/ra_tiff.dir/ra_tiff.c.o
[100%] Building C object src/px/MakeFiles/normtiff.dir/normtiff.c.o
[100%] Linking C executable ../bin/normtiff
[100%] Linking C executable ../bin/ra_tiff
[100%] Built target normtiff
[100%] Built target ra_tiff
Uploading artifacts...
build/: found 3632 matching files
Uploading artifacts to coordinator... ok
Job succeeded
  
```

The right sidebar provides job details: 'build:macclang' with a duration of 2 minutes 13 seconds and a timeout of 16h 39m. The runner is 'macOS 10.12 - aswindle-31874s (#34)'. The job artifacts section indicates that artifacts were removed 47 minutes ago. The commit is 'a46558bb'. A pipeline dropdown shows 'Pipeline #2449 from master' with a 'build' job selected. Other jobs in the pipeline include 'build:windows', 'build:ubuntu', and 'build:mac:gcc'.

# GitLabCI - Tests Passed, Make new release?



The screenshot shows a GitLab CI job page for the project 'Radiance'. The job is named 'test:windows' and has a status of 'passed'. The job was triggered about 2 hours ago by a bot. The job details are as follows:

- Duration: 46 seconds
- Timeout: 16h 39m (from project)
- Runner: Windows 10 - aswindle-31874s (#35)
- Tags: windows
- Commit: a46558bb
- Added missing file from last commit
- Pipeline: #2449 from master
- Test: test

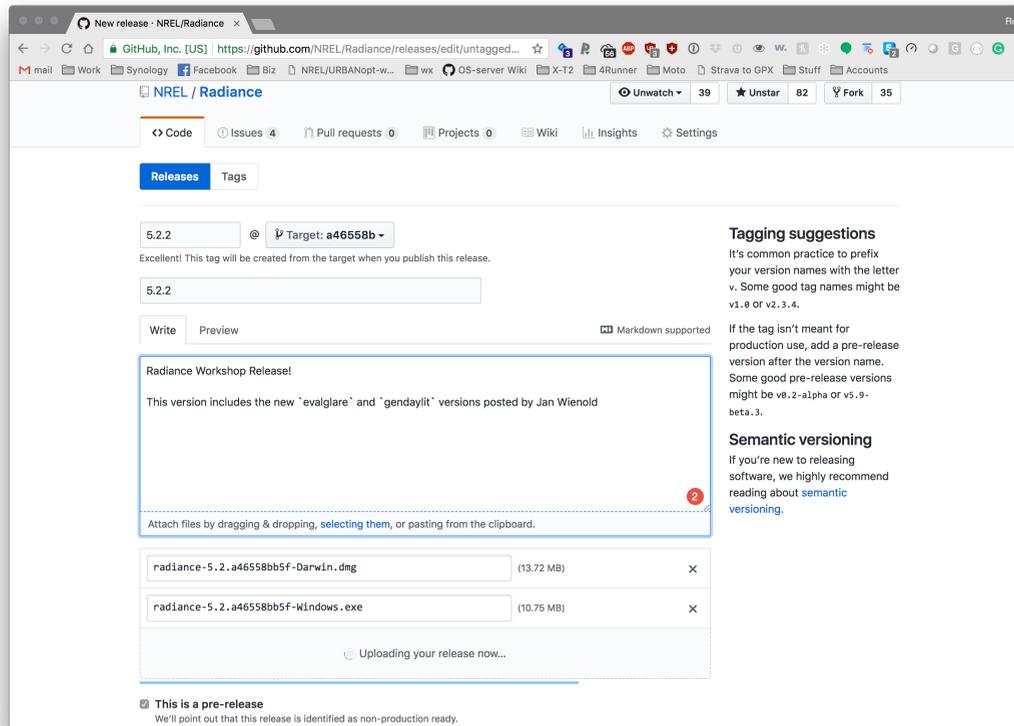
The job log shows the following steps:

```
Running with gitlab-runner 10.5.0 (80b03db9)
on aswindle-W10M c1016289
Using Shell executor...
Running on ASWINDLE-W10M...
fetching changes...
Removing build/
HEAD is now at a46558bb Added missing file from last commit
Checking out a46558bb as master...
Skipping git submodules setup
Downloading artifacts for build:windows (14804)...
Downloading artifacts from coordinator... ok id=14800 responseStatus=200 OK token=jlBPg5r3
$ cd build
$ "C:\Program Files\Make\bin\ctest.exe"
Test project C:/GitLab-Runner/builds/c1016289/0/NREL/Radiance/build
Start 1: test_setup
1/7 Test #1: test_setup ..... Passed 0.08 sec
Start 2: test_rtrace
2/7 Test #2: test_rtrace ..... Passed 0.08 sec
Start 3: test_rpict
3/7 Test #3: test_rpict ..... Passed 2.95 sec
Start 4: test_getinfo
4/7 Test #4: test_getinfo ..... Passed 0.10 sec
Start 5: test_pcond
5/7 Test #5: test_pcond ..... Passed 1.26 sec
Start 6: test_evalglare
6/7 Test #6: test_evalglare ..... Passed 7.21 sec
Start 7: test_falsecolor
7/7 Test #7: test_falsecolor ..... Passed 7.74 sec

100% tests passed, 0 tests failed out of 7

Total Test time (real) = 19.46 sec
Job succeeded
```

# New 'Workshop Edition' Release



The screenshot shows the GitHub interface for creating a new release. The repository is NREL/Radiance. The release name is 5.2.2, and the target commit is a46558bb. The release description is: "Radiance Workshop Release! This version includes the new 'evalgiare' and 'gendaylit' versions posted by Jan Wienold". Two files are attached: radiance-5.2.a46558bb5f-Darwin.dmg (13.72 MB) and radiance-5.2.a46558bb5f-Windows.exe (10.75 MB). The release is marked as a pre-release.

New release - NREL/Radiance

GitHub, Inc. [US] | <https://github.com/NREL/Radiance/releases/edit/untagged...>

NREL / Radiance

Unwatch 39 | Unstar 82 | Fork 35

Code | Issues 4 | Pull requests 0 | Projects 0 | Wiki | Insights | Settings

Releases | Tags

5.2.2 @ Target: a46558bb

Excellent! This tag will be created from the target when you publish this release.

5.2.2

Write | Preview | Markdown supported

**Radiance Workshop Release!**

This version includes the new "evalgiare" and "gendaylit" versions posted by Jan Wienold

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

radiance-5.2.a46558bb5f-Darwin.dmg (13.72 MB) x

radiance-5.2.a46558bb5f-Windows.exe (10.75 MB) x

Uploading your release now...

This is a pre-release  
We'll point out that this release is identified as non-production ready.

**Tagging suggestions**  
It's common practice to prefix your version names with the letter v. Some good tag names might be v1.0 or v2.3.4.

If the tag isn't meant for production use, add a pre-release version after the version name. Some good pre-release versions might be v0.2-alpha or v5.9-beta.3.

**Semantic versioning**  
If you're new to releasing software, we highly recommend reading about [semantic versioning](#).

## Next Steps

### More Testing

- Currently the CMake Test suite is a handful of regression, and simple sanity tests
- Need more tests, more coverage
- There is some DOE funding for this in FY19!

### Improved Packages

- Still missing some supplemental files
  - `picture.cal` `dirt.cal` (and more!)
- Better control of setting `PATH`, `RAYPATH`, other env vars (Mac)
- Build pdfs of manpages for Windows users
  - Mac/Linux get manpages installed to `[radiance_install_dir]/man` automatically

## Next Steps (Cont'd.)

### More Package Options

- Better linux package options
  - Current "installer" is just a tarball of 64-bit executables and libs built on Ubuntu
  - Randolph Fritz working on a MacPorts "port" for Radiance
  - Other packages? (rpm? brew? what? WHAT!?)

### More Automation

- Automatic Package Posting?
- Iteration builds (i.e. last successful HEAD commit always available)?

### Documentation

- Help individual users make their own builds
- User-level repository

# Bonus Slide

## Your own Private Repo

1. Clone the NREL Git repository:

```
$ git clone https://github.com/NREL/Radiance.git
```

2. Refresh your local copy whenever you want:

```
$ git cvsimport -o master -v -d \  
:pserver:anonymous@radiance-online.org:/home/cvsd/radiance \  
-A ../authors.txt -a ray
```

authors.txt = maps CSH logins to GitHub usernames

## Bonus Slide 2

### What's a test?

```
execute_process(  
  WORKING_DIRECTORY ${test_output_dir}  
  COMMAND evalglare${CMAKE_EXECUTABLE_SUFFIX} -L 400 500 0.6 1.2 -B 0.4 \  
  -b 2500 -d -c outputimage.hdr ${resources_dir}/evalglare/testimage.hdr  
  OUTPUT_FILE evalglare_out.txt  
  RESULT_VARIABLE res  
)  
  
file(READ ${test_output_dir}/evalglare_out.txt test_output)  
if(test_output MATCHES "band:band_omega,band_av_lum,band_median_lum, \  
band_std_lum,band_perc_75,band_perc_95,band_lum_min,band_lum_max: \  
2.423399 2353.496003 1339.703125 2218.441248 3993.937463 5750.375098 \  
68.348633 37142.499489")  
  message(STATUS "passed")  
else()  
  message(STATUS "failed")  
endif()
```

# Questions?

[robert.guglielmetti@nrel.gov](mailto:robert.guglielmetti@nrel.gov)

[www.nrel.gov/buildings](http://www.nrel.gov/buildings)

[github.com/NREL/Radiance](https://github.com/NREL/Radiance)

[rob.guglielmetti@gmail.com](mailto:rob.guglielmetti@gmail.com)

[www.luminousbits.com](http://www.luminousbits.com)

[thank you.(!)]

# Questions?

[robert.guglielmetti@nrel.gov](mailto:robert.guglielmetti@nrel.gov)

[www.nrel.gov/buildings](http://www.nrel.gov/buildings)

[github.com/NREL/Radiance](https://github.com/NREL/Radiance)

[rob.guglielmetti@gmail.com](mailto:rob.guglielmetti@gmail.com)

[www.luminousbits.com](http://www.luminousbits.com)

[thank you.(!)]

