

# A Crash Course for **SU2** Developers

code

**Tim Albring**, Developer Director

1st SU2 Conference 2020

# So, you want to be an SU2 developer?

It's easier than you might think.



We leverage **standard development processes** and the **latest tools for open-source projects**.

# Disclaimer - What this talk is not

- *A programming tutorial*
- *A guide on how to install and build SU2*
- *A source-code walkthrough*
- *A professional software engineering lecture*
- *Only for experienced developers*
- *Only for new developers*



Git/Branching

Development

Pull Request

Testing

Release

su2code / SU2

Unwatch 167 Unstar 595 Fork 494

<> Code Issues 32 Pull requests

SU2: An Open-Source Suite for Multiphysics Simulation

cfd c-plus-plus simulation optimization

11,664 commits 209 branches

Branch: master New pull request

talbring Merge pull request #1007 from su2code/master

github Common QuickStart SU2\_CFD

added error message in case steady outputfield was not found 3d444d

fixed failed build

fixed issue

adapted not

small bug i

updated tes

Merge branch

configured

Merge branch

further con

pcarruscag approved these changes 20 days ago

pcarruscag left a comment

LGTM

pr-triage 20 days ago

pr-triage 17 days ago

pcarruscag 17 days ago

16 checks passed

Fixed a bug for the interpolation weights of a tetrahedron

codefactor.io

Regression and Unit Testing on: pull\_request

- ✓ Build SU2 (BaseMPI)
- ✓ Build SU2 (ReverseMPI)
- ✓ Build SU2 (ForwardMPI)
- ✓ Build SU2 (BaseNoMPI)
- ✓ Build SU2 (ReverseNoMPI)
- ✓ Build SU2 (ForwardNoMPI)
- ✓ Build SU2 (BaseOMP)
- ✓ Regression Tests (tutorial)
- ✓ Regression Tests (parallel)
- ✓ Regression Tests (parallel)
- ✓ Regression Tests (serial)
- ✓ Regression Tests (serial)
- ✓ Regression Tests (hybrid)
- ✓ Unit Tests (test\_driver)
- ✓ Unit Tests (test\_driver)
- ✓ Unit Tests (test\_driver)

Regression and Unit Testing succeeded 6 hours ago

Set up job

Pull su2code

Cache Objects

Build

Upload Binaries

Post Cache

Complete

SU2 version 7.0.5 "Blackbird"

github-actions released this 6 days ago

Changes

Experimental Features

- Multiline config option values and Paraview vtu default file format @talbring (#996)
- Mixed precision linear algebra @pcarruscag (#989)

Bug Fixes

- adding reference for distortion and uniformity parameters @auzbaig (#985)
- Restart Features for some python scripts @ScSteffen (#964)
- Adds Grid Velocity Term in the Euler/Symmetry BC @economon (#1004)
- Restart and output fix for (very) large meshes @GomerOfDoom (#980)
- Multigrid and preprocessing efficiency tweaks @pcarruscag (#963)
- Fix spelling error in CSinglezoneDriver.cpp. Add missing comment for iteration\_structure.hpp @leminhman0312 (#987)
- Update Xcode and one bug fix @tlaui (#981)

Maintenance

- Adds Missing Const Specifiers to Member Functions @economon (#993)
- Move FEM files @pcarruscag (#1001)

Git/Branching

Development

Pull Request

Testing

Release

su2code / SU2

Unwatch 167 Unstar 595 Fork 494

<> Code Issues 32 Pull requests

SU2: An Open-Source Suite for Multiphysics Simulation

cfd c-plus-plus simulation optimization

11,664 commits 209 branches

Branch: master New pull request

talbring Merge pull request #1007 from su2code/master

github Common QuickStart SU2\_CFD

added error message in case steady outputfield was not found 3d444d

fixed failed build

fixed issue

adapted not

small bug i

updated tes

Merge branch

configured

Merge branch

further con

## Let's start from the beginning ...

pcarruscag approved these changes 20 days ago

pcarruscag left a comment

LGTM

pr-triage 20 days ago

pr-triage 17 days ago

pcarruscag 17 days ago

16 checks passed

Fixed a bug for the interpolation weights of a tet a100c9c

WIP

codefactor.io

Regression and Unit Testing on: pull\_request

- ✓ Build SU2 (BaseMPI)
- ✓ Build SU2 (ReverseMPI)
- ✓ Build SU2 (ForwardMPI)
- ✓ Build SU2 (BaseNoMPI)
- ✓ Build SU2 (ReverseNoMPI)
- ✓ Build SU2 (ForwardNoMPI)
- ✓ Build SU2 (BaseOMP)
- ✓ Regression Tests (tutorial)
- ✓ Regression Tests (parallel)
- ✓ Regression Tests (parallel)
- ✓ Regression Tests (serial)
- ✓ Regression Tests (serial)
- ✓ Regression Tests (hybrid)
- ✓ Unit Tests (test\_driver)
- ✓ Unit Tests (test\_driver)
- ✓ Unit Tests (test\_driver)

Regression and Unit Testing succeeded 6 hours ago

Set up job

Pull su2code

Cache Objects

Build

Upload Binaries

Post Cache

Complete

SU2 version 7.0.5 "Blackbird"

github-actions released this 6 days ago

### Changes

#### Experimental Features

- Multiline config option values and Paraview vtu default file format @talbring (#996)
- Mixed precision linear algebra @pcarruscag (#989)

#### Bug Fixes

- adding reference for distortion and uniformity parameters @auzbaig (#985)
- Restart Features for some python scripts @ScSteffen (#964)
- Adds Grid Velocity Term in the Euler/Symmetry BC @economon (#1004)
- Restart and output fix for (very) large meshes @GomerOfDoom (#980)
- Multigrid and preprocessing efficiency tweaks @pcarruscag (#963)
- Fix spelling error in CSinglezoneDriver.cpp. Add missing comment for iteration\_structure.hpp @leminhman0312 (#987)
- Update Xcode and one bug fix @tlaui (#981)

#### Maintenance

- Adds Missing Const Specifiers to Member Functions @economon (#993)
- Move FEM files @pcarruscag (#1001)

Git/Branching

Development

Pull Request

Testing

Release

su2code / SU2

Unwatch 168 Unstar 602 Fork 497

<> Code Issues 30 Pull requests 14 Actions Projects 4 Security 0 Insights Settings

SU2: An Open-Source Suite for Multiphysics Simulation and Design <https://su2code.github.io> Edit

cfd c-plus-plus simulation optimization python opensource physics flow fluid fluid-dynamics hpc Manage topics

11,664 commits 208 branches 1 package 43 releases 1 environment 72 contributors LGPL-2.1

Branch: master New pull request Create new file Upload files Find file Clone or download

talbring Merge pull request #1007 from su2code/release\_v7.0.5

|            |                                      |
|------------|--------------------------------------|
| .github    | put mixed precision under regression |
| Common     | Changing version number to 7.0.5     |
| QuickStart | Changing version number to 7.0.5     |
| SU2_CFD    | Changing version number to 7.0.5     |
| SU2_DEF    | Changing version number to 7.0.5     |
| SU2_DOT    | Changing version number to 7.0.5     |
| SU2_GEO    | Changing version number to 7.0.5     |
| SU2_IDE    | update xcode - finished              |

Clone with HTTPS Use Git or checkout with SVN using the web URL.

<https://github.com/su2code/SU2.git>

~~Download ZIP~~

Please don't



Git/Branching

Development

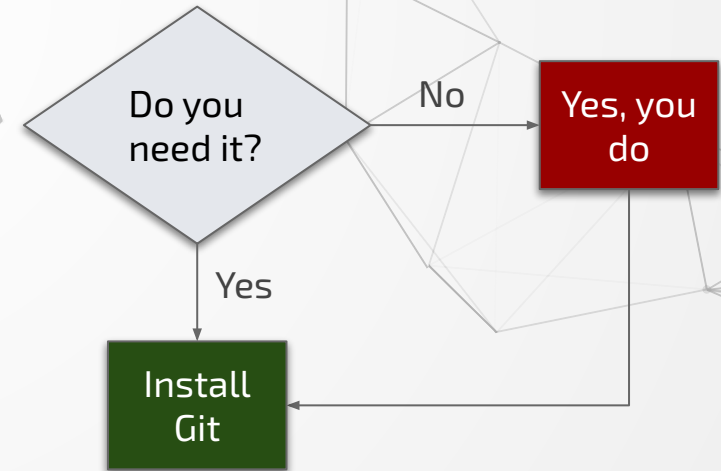
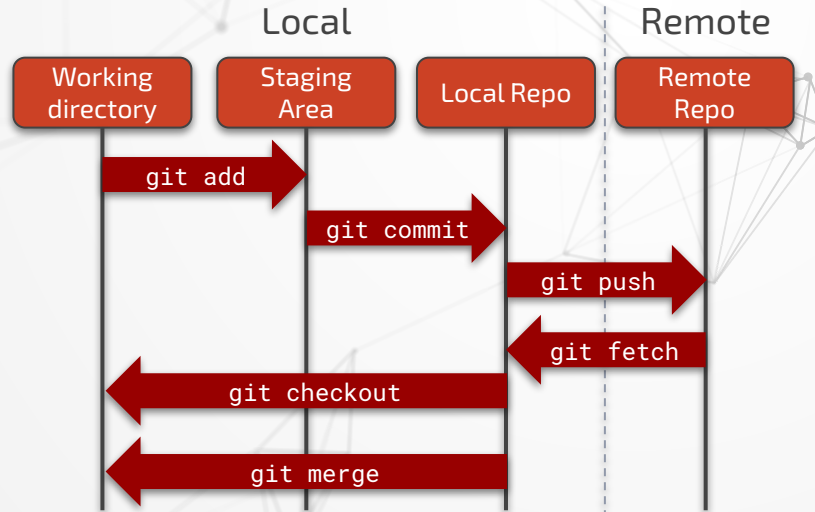
Pull Request

Testing

Release

# What are Git and Github?

- Git is a **version control system** to keep track of changes to files over time
- Github is a website that hosts Git **repositories** online
- Each change made to the repository code is a '**commit**', often with a supporting comment about the change that was made



Git/Branching

Development

Pull Request

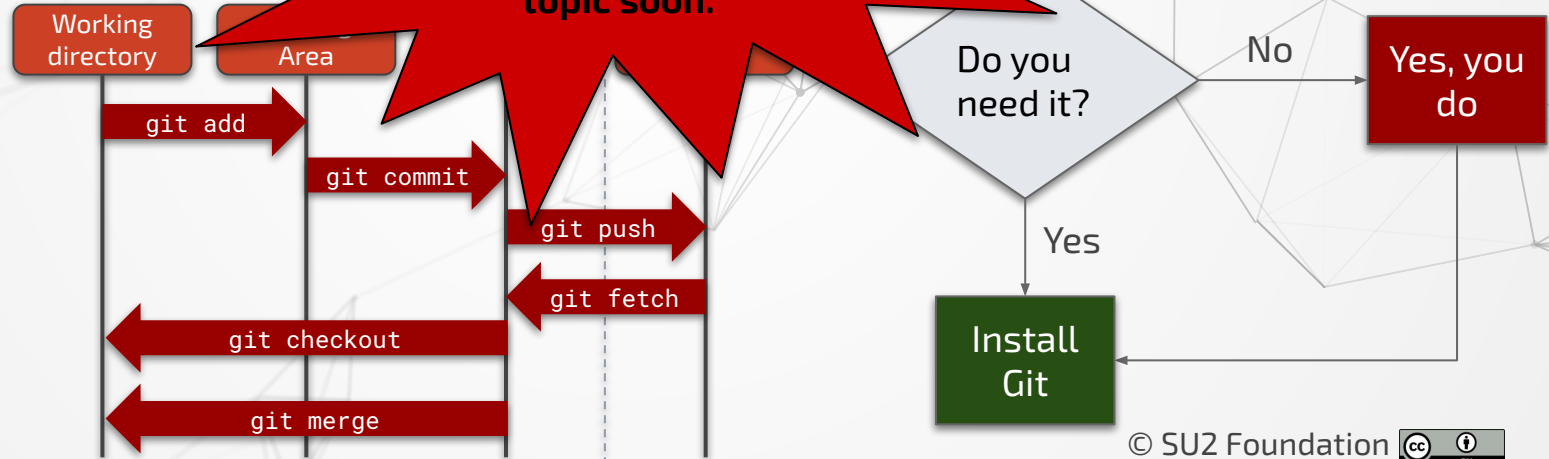
Testing

Release

## What are Git and Github?

- Git is a **version control system** to keep track of changes to files over time
- Github is a website that hosts **repositories** online
- Each change made to the repository is called a **commit** with a supporting comment about the change that was made

**If there is enough interest, we can offer an online course on this topic soon.**





Git/Branching

Development

Pull Request

Testing

Release

Master

Develop

Feature

Feature

Tag

v0.1

v0.2

v1.0

Branch

Commit

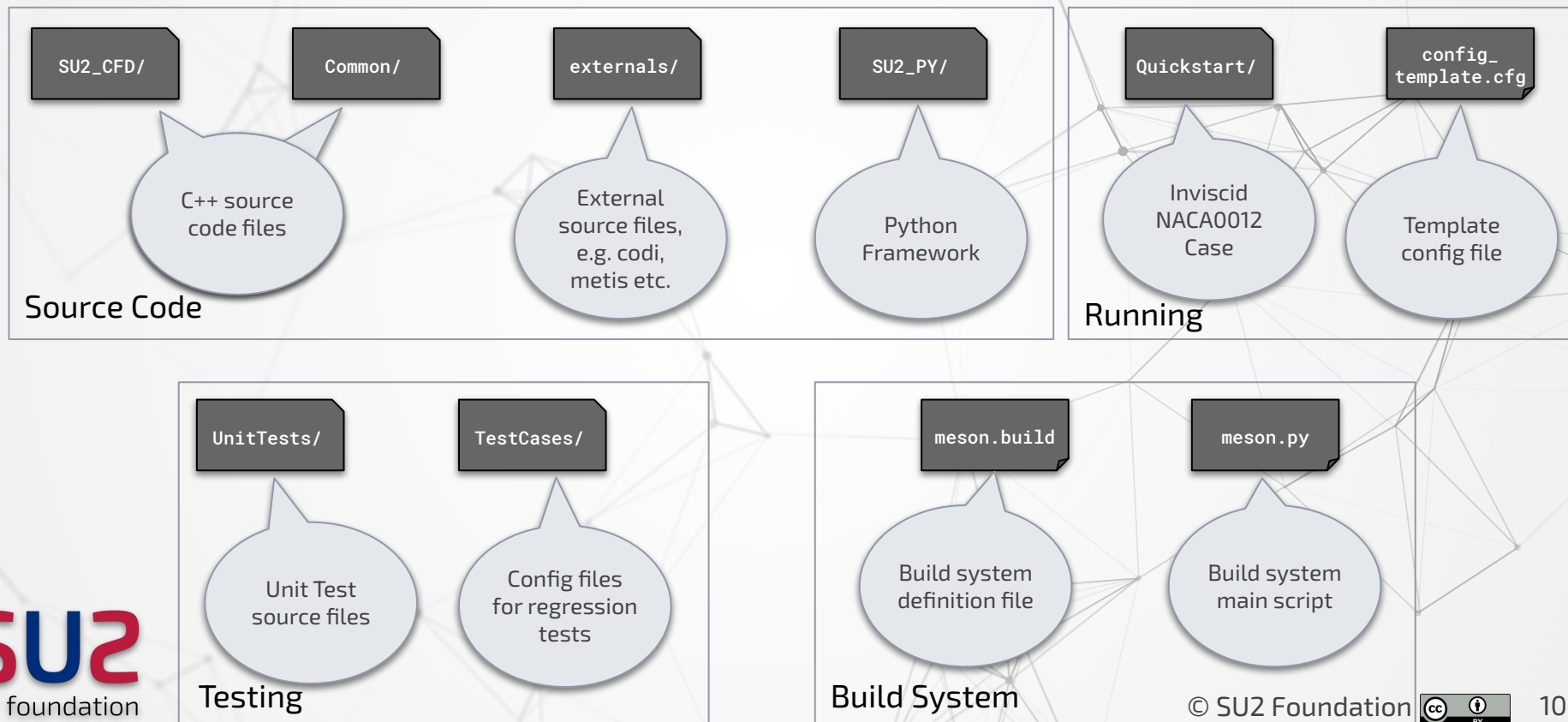
Git/Branching

Development

Pull Request

Testing

Release



Git/Branching

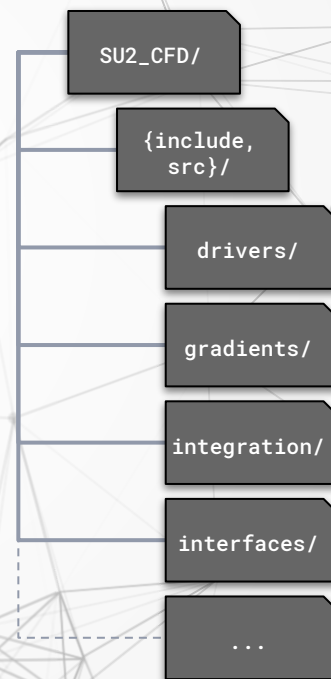
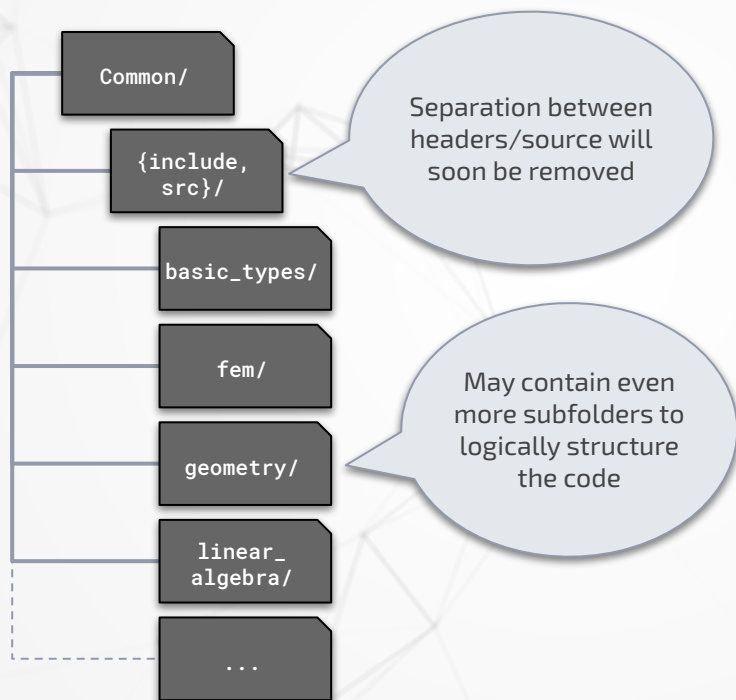
Development

Pull Request

Testing

Release

## Structure of the Source Code



Git/Branching

Development

Pull Request

Testing

Release

Before starting to code - these are the topics you should get familiar with in C++:

- Classes and polymorphism
- Virtual functions
- Pointers
- *Message Passing Interface (MPI)*

Plus, some C++11 features like **auto**, **range-based loops**, **lambda functions** can help

Git/Branching

Development

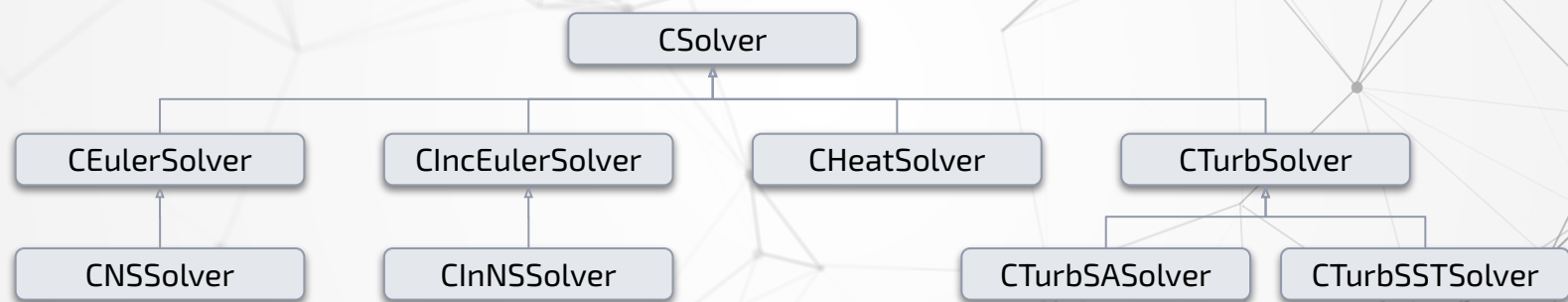
Pull Request

Testing

Release

C++ class abstractions encourage code reuse and data encapsulation ensures you can make localized changes easily.

Example:



Git/Branching

Development

Pull Request

Testing

Release

High-level classes

CSinglezoneDriver

Cliteration

CIntegration

CSolver

CIntegration

CSolver

CIntegration

CSolver

COutput

Git/Branching

Development

Pull Request

Testing

Release

## High-level classes - Example

CSinglezoneDriver

CFluidIteration

CMultiGridIntegration

CIncNSSolver

Main solver,  
based on  
SOLVER  
option

CSingleGridIntegration

CTurbSASolver

Sub solver

CSingleGridIntegration

CRadP1Solver

Sub solver

CFlowIncOutput

**Pro Tip:** For most of the classes there exist **Factory methods** to create them (e.g. CSolverFactory)



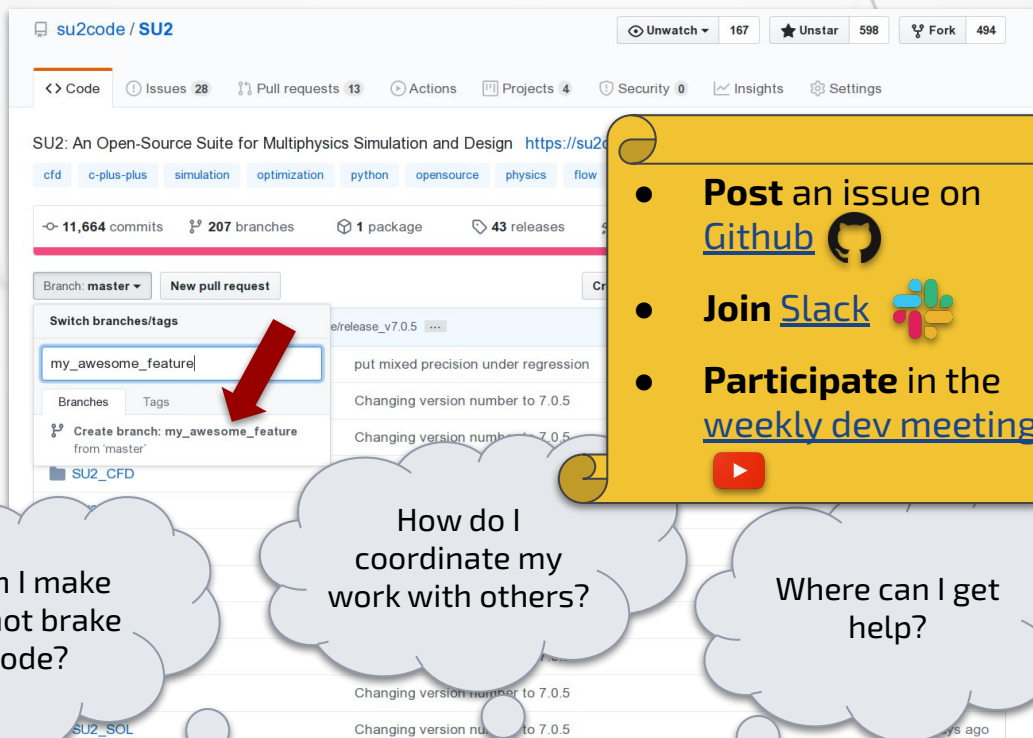
Git/Branching

Development

Pull Request

Testing

Release



How can I make  
sure to not brake  
the code?

How do I  
coordinate my  
work with others?

Where can I get  
help?

Create your branch!

Git/Branching

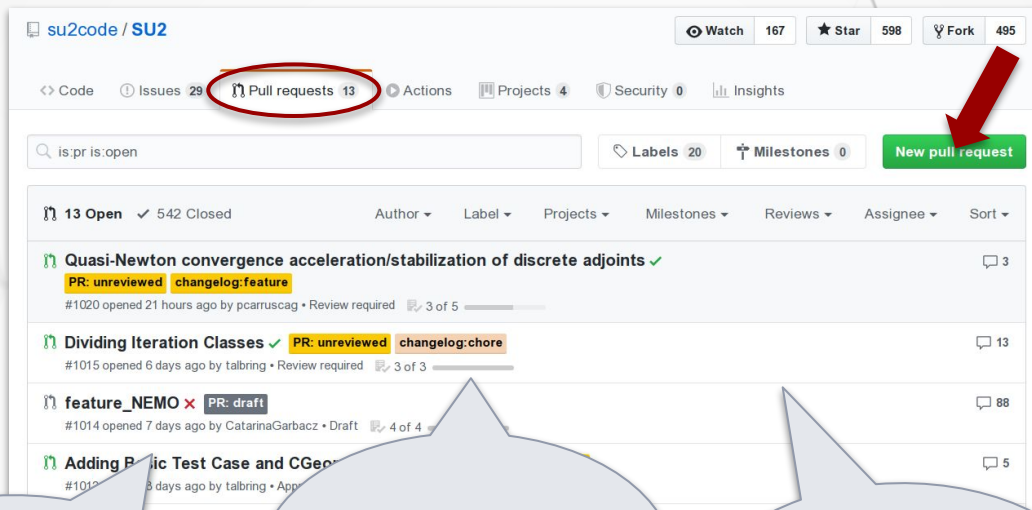
Development

Pull Request

Testing

Release

A PR is a **request** to the project to pull in your contribution. Can be from an internal branch or from an external fork.



Make sure to not brake the code!

Coordinate your work with others!

Get help!

And create a pull request!

Git/Branching

Development

Pull Request

Testing

Release

su2code / SU2

Unwatch 16

<> Code Issues 29 Pull requests 13 Actions Projects 4 Security 0

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across](#)

base: **development** ← compare: **feature\_logging** × **Can't automatically merge.** Don't worry, you can still create the pull request.



My Awesome Feature

Write Preview

H B I ≡ <> ⌂ ☑ @ ↩ ↶

## Proposed Changes

\*Give a brief overview of your contribution here in a few sentences.\*

## Related Work

\*Resolve any issues (bug fix or feature request), note any related PRs, or mention interactions with the work of others, if any.\*

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

Create pull request

✓ Create pull request

Open a pull request that is ready for review

Create draft pull request

Cannot be merged until marked ready for review

We strongly suggest to open a PR as early as possible! Even if it is not ready!

Submit the PR to the **develop** branch

Fill out the PR template questions that guide you along your way.

Use a **descriptive title**

Choose a **label** so that your feature is added to the changes for the release

If your work is ready choose this

Choose this if your work is **not** ready **yet**

Git/Branching

Development

Pull Request

Testing

Release

## Benefits of opening a (draft) Pull Request

- ✓ **Coordination:** Let others know what you are working on
- ✓ **Feedback and help:** Experienced developers can help you and can guide you during the development process
- ✓ **Automated Building and Testing:** You instantly know if something is broken or results have changed in a commit
- ✓ **Faster review process:** Your PR is likely to be merged in sooner

Git/Branching

Development

Pull Request

Testing

Release

economom approved these changes 7 days ago [View changes](#)

economom left a comment [Member](#)

LGTM, just small comments

If you have time to add tests for the readers, that would be a big bonus 🍌

```
UnitTests/Common/geometry/CGeometry_test.cpp  Outdated
137 + CHECK(TestCase->geometry->vertex[3][2]->GetNormal())[1] == -0.0625);
138 + CHECK(TestCase->geometry->vertex[5][3]->GetNormal())[2] == 0.03125);
139 +
140 + }
```

economom 7 days ago [Member](#)

Add end lines to these files

[Reply...](#)

[Resolve conversation](#)

```
UnitTests/Common/geometry/CGeometry_test.cpp
...  ...  @@ -0,0 +1,140 @@
1 + #include "catch.hpp"
```

economom 7 days ago [Member](#)

Add the typical file header info to these files

[Reply...](#)

[Resolve conversation](#)

Don't see review comments as a criticism of your work. It's to help you and us to maintain a great product!

And trust me, you'll learn a lot in that process.

PR needs approval from **at least two** people from the community

All checks must pass

**Changes approved** [Hide all reviewers](#)

2 approving reviews by reviewers with write access. [Learn more.](#)

2 approvals

economom approved these changes

pcarruscag approved these changes

**All checks have passed** [Hide all checks](#)

18 successful checks

- Regression and Unit Testing / Build SU2 (BaseMPI) (pull\_request) ... [Required](#) [Details](#)
- Regression and Unit Testing / Build SU2 (ReverseMPI) (pull\_request) ... [Required](#) [Details](#)
- Regression and Unit Testing / Build SU2 (ForwardMPI) (pull\_request) ... [Required](#) [Details](#)
- Regression and Unit Testing / Build SU2 (BaseNoMPI) (pull\_request) ... [Required](#) [Details](#)
- Regression and Unit Testing / Build SU2 (ReverseNoMPI) (pull\_request) ... [Required](#) [Details](#)
- Regression and Unit Testing / Build SU2 (ForwardNoMPI) (pull\_request) ... [Required](#) [Details](#)

**This branch is out-of-date with the base branch**

Merge the latest changes from 'devel' into this branch.

This merge commit will be associated with tim.albring@scicomp.uni-kl.de.

[Update branch](#)

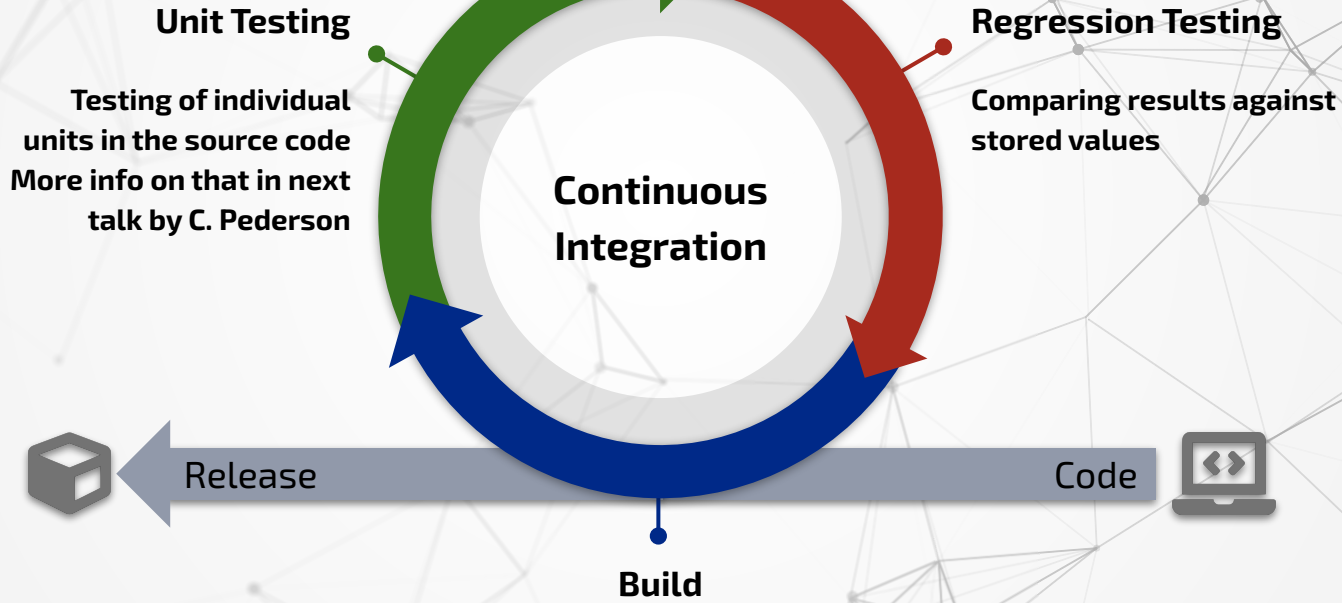
Git/Branching

Development

Pull Request

Testing

Release



Git/Branching

Development

Pull Request

Testing

Release

# GitHub Actions



On **every push** to a **branch**  
opened as a (draft) **Pull Request**:

1. The code is built
2. Regression tests run
3. Unit tests are executed

The screenshot shows a GitHub repository for 'su2code / SU2'. The 'Actions' tab is selected, displaying a workflow named 'fix nOMPI build'. The workflow is triggered on a pull request. The job list on the left includes:

- Build SU2 (BaseMPI)
- Build SU2 (ReverseMPI)
- Build SU2 (ForwardMPI)
- Build SU2 (BaseNoMPI)
- Build SU2 (ReverseNoMPI)
- Build SU2 (ForwardNoMPI)
- Build SU2 (BaseOMP)
- Regression Tests (tutorial...)
- Regression Tests (paralle...)
- Regression Tests (paralle...)
- Regression Tests (serial\_...)
- Regression Tests (serial\_...)
- Regression Tests (hybrid...)
- Unit Tests (test\_driver)
- Unit Tests (test\_driver\_AD)
- Unit Tests (test\_driver\_DD)

The right panel shows the details of the 'Regression and Unit Testing / Build SU2 (BaseMPI)' job, which succeeded 14 hours ago. The job steps are:

- Set up job (2s)
- Pull su2code/build-su2:20191105 (15s)
- Cache Object Files (57s)
- Build (10m 11s)
- Upload Binaries (14s)
- Post Cache Object Files (14s)
- Complete job (0s)

A speech bubble indicates: 'If any of the steps fail, you'll receive an email'.



Git/Branching

Development

Pull Request

Testing

Release

Add a regression test to protect your features

```
# NACA0012
naca0012 = TestCase('naca0012')
naca0012.cfg_dir = "euler/naca0012"
naca0012.cfg_file = "inv_NACA0012_Roe.cfg"
naca0012.test_iter = 20
naca0012.test_vals = [-4.047448, -3.538057, 0.338691, 0.023131] #last 4 columns
naca0012.su2_exec = "SU2_CFD"
naca0012.timeout = 1600
naca0012.tol = 0.00001
test_list.append(naca0012)
```

If the computed values after your commit don't match these values, the test will fail

1. Add a new test case to **serial\_regression.py**, **parallel\_regression.py**, etc. Use others as a guide. See NACA 0012 example.

Git/Branching

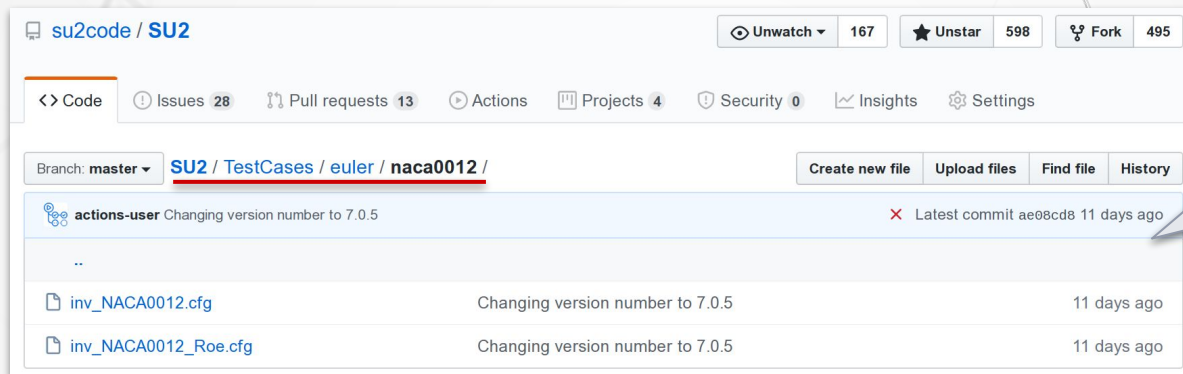
Development

Pull Request

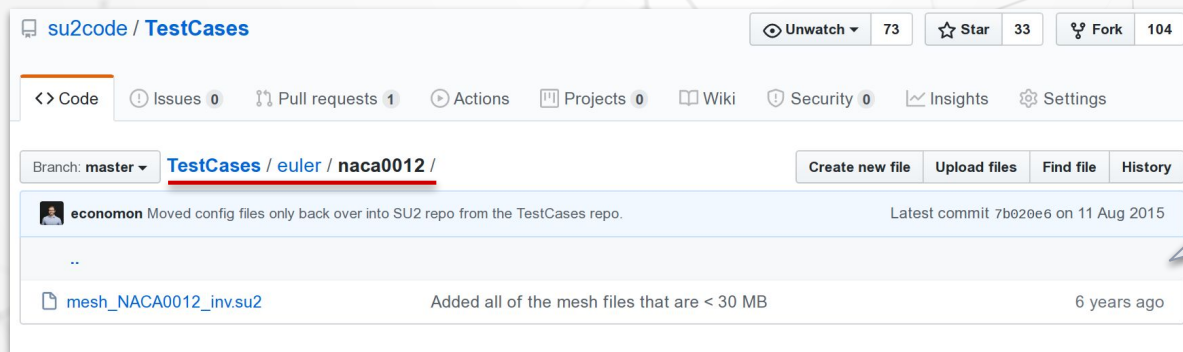
Testing

Release

2. Put the config file and any supporting data in the corresponding locations.



Lighter weight,  
more frequently  
updated files go in  
code repo.



Larger, more static  
files that support  
the tests go in the  
TestCases repo.

Git/Branching

Development

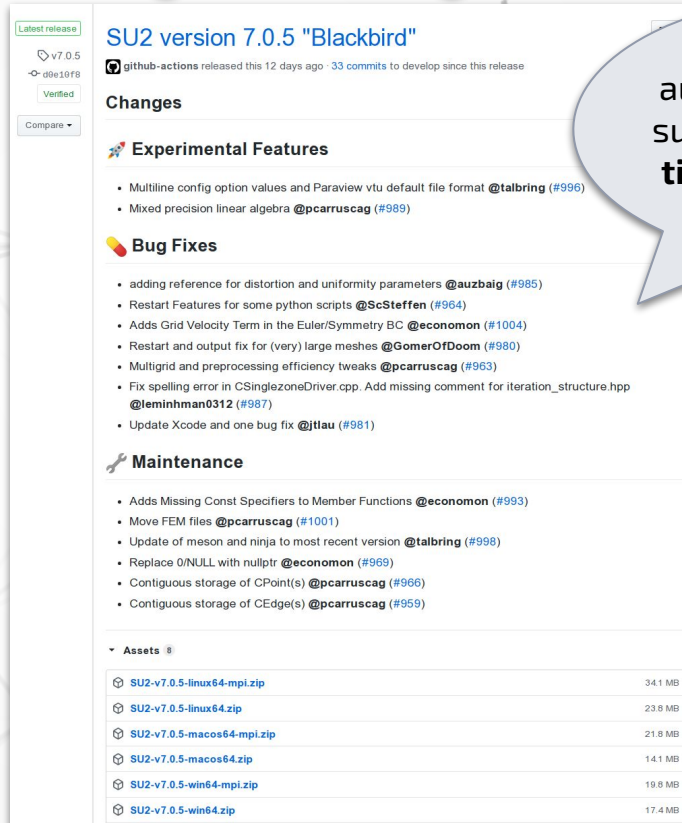
Pull Request

Testing

Release

Frequently a new version **with your contribution** will be released, which means

1. develop branch is moved to master
2. A new tag is created
3. Binaries are created
4. Release notes are published
5. New version is advertised



**SU2 version 7.0.5 "Blackbird"**

github-actions released this 12 days ago · 33 commits to develop since this release

**Changes**

**Experimental Features**

- Multiline config option values and Paraview vtk default file format @talbring (#996)
- Mixed precision linear algebra @pcarruscag (#989)

**Bug Fixes**

- adding reference for distortion and uniformity parameters @auzbaig (#985)
- Restart Features for some python scripts @ScSteffen (#964)
- Adds Grid Velocity Term in the Euler/Symmetry BC @economon (#1004)
- Restart and output fix for (very) large meshes @GomerOfDoom (#980)
- Multigrid and preprocessing efficiency tweaks @pcarruscag (#963)
- Fix spelling error in CSinglezoneDriver.cpp. Add missing comment for iteration\_structure.hpp @leminhman0312 (#987)
- Update Xcode and one bug fix @jltau (#981)

**Maintenance**

- Adds Missing Const Specifiers to Member Functions @economon (#993)
- Move FEM files @pcarruscag (#1001)
- Update of meson and ninja to most recent version @talbring (#998)
- Replace O/NULL with nullptr @economon (#969)
- Contiguous storage of CPoint(s) @pcarruscag (#966)
- Contiguous storage of CEdge(s) @pcarruscag (#959)

**Assets**

|                            |         |
|----------------------------|---------|
| SU2-v7.0.5-linux64-mpi.zip | 34.1 MB |
| SU2-v7.0.5-linux64.zip     | 23.8 MB |
| SU2-v7.0.5-macos64-mpi.zip | 21.8 MB |
| SU2-v7.0.5-macos64.zip     | 14.1 MB |
| SU2-v7.0.5-win64-mpi.zip   | 19.8 MB |
| SU2-v7.0.5-win64.zip       | 17.4 MB |

Everything happens automatically, so make sure to choose a **proper title** and **label** for your PR

Git/Branching

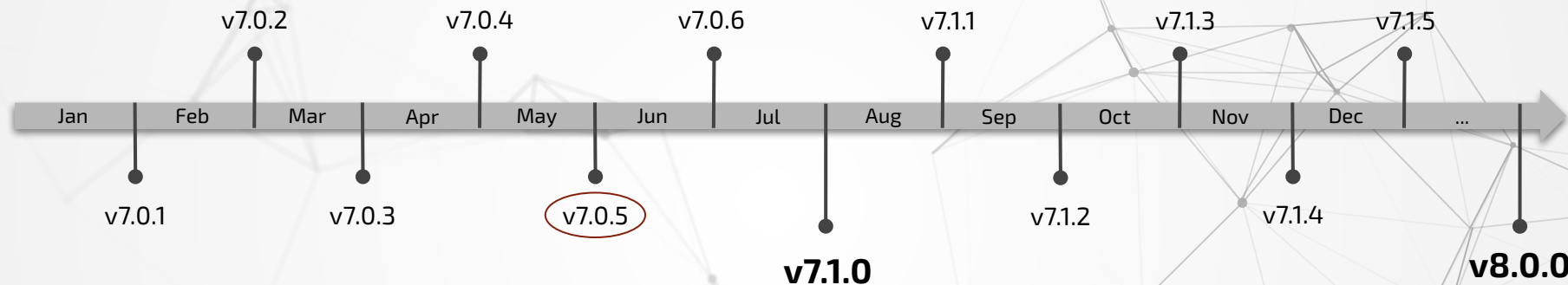
Development

Pull Request

Testing

Release

## Release Schedule



**Maintenance releases** every month

**Minor releases** every 6 months (may vary)

**Major releases** every x year(s) (depends)

Goes along with updates in **documentation** and may break **backward compatibility** of config files

Git/Branching

Development

Pull Request

Testing

Release

## SU2 Development Survival Guide

1. **Clone main repository:** `$ git clone https://github.com/su2code/SU2.git`
2. **Create new feature branch** for your development work. Work on this branch in the repo.
3. **Open a draft PR** so that other developers immediately see what you are working on. Also you can leverage the testing system! Note that any new commits will appear right there in the open PR and will kick-off the tests again.
4. **Work on your feature!** Please mind white space issues, compiler warnings, and match the SU2 style -> Make use of [clang-format](#).
5. **Merge in the remote develop branch into your branch as frequent as possible.** This ensures that, when the time comes and your feature is finished, it will be easy to merge your contribution into develop, as you will have solved any conflicts on your side.
6. Once you feel your feature is finished, **mark your PR as "Ready to Review"**.
7. **Get reviews and engage with the community concerning your contribution.** Fix problems in your branch or address any feedback on the message boards.
8. Once the reviewers approve and all tests pass, **the community will merge in your work.**
9. Celebrate your contribution and proudly introduce yourself as an **SU2 developer** at your social engagements.

# Pretty easy, right?

We have set up safety nets and removed overhead wherever possible.

So, try crazy ideas in your branches. **Don't be afraid to make big changes** that push the boundaries of the code.

The community and infrastructure will be there to help you. This is how we make progress.

