PROJECT CHRONO

Git and GitHub primer

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

UNIVERSITÀ DEGLI STUDI DI PARMA

What is version control?

- Aka revision control, source control
- Version control is the management and tracking of changes to source code, documents, data, etc.
- Allows collaborative development
- Keeps track of who made a change, when the change was made, and what the change was
- Permits reverting any change and rolling back to a previous state

- Many systems available: CVS, Subversion (SVN), Perforce, git, Mercurial,…

What is git?

- Distributed revision control system
  - Speed
  - Data integrity
  - Distributed, non-linear work flows
- Created in 2005 by Linus Torvalds to support the Linux kernel development

- Main characteristics:
  - the entire code and history is kept on the client (user) machine
  - users can work (make changes to code) even without internet connection
  - internet connection required only for pushing and pulling from remote server (remote repository)

Git basics – commits

- Git keeps track of code history in snapshots
  - record of what all files look like at a given point in time
- User decides when to take a snapshot (commit) and what files should be included
- Allows going back and visiting any past snapshot
  - later snapshots are not lost
- A project is made out of a series of commits
- Each commit contains:
  1. information on how the files changed from previous commit (diff)
  2. a reference to the previous commit (parent commit)
  3. a hash code name

Git basics – repositories

- A repository (or ‘repo’) is a collection of all the files and their commit history
  - contains all commits
  - can be local or remote
- Copying a repository from a remote server is called cloning
- Cloning allows teams to develop collaboratively
- Pulling: downloading commits that do not exist on the local machine from a remote repository
- Pushing: adding local changes (commits) to a remote repository

Git basics – branches

- All commits in a repository live in some branch
- The main branch in a repository is called the master branch
- A project can have many branches
  - For example, in a project that follows GitFlow, will have a master branch, a develop branch, feature branches, hotfix branches, release branches
- Branches allow maintaining parallel and separate development tracks in a single project
- Development tracks can be
  - branched off
  - merged

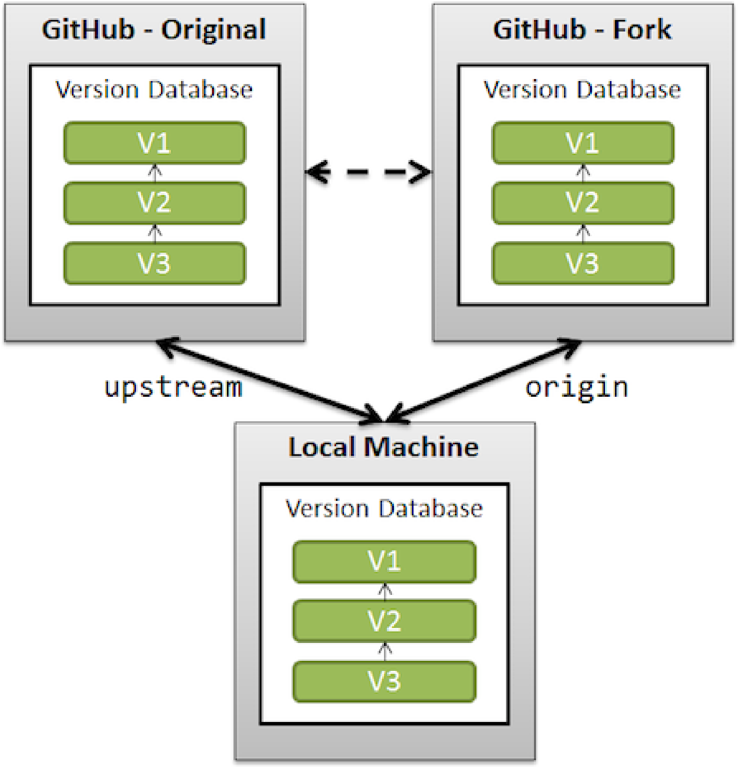
What is GitHub?

- Largest web-based git repository hosting service https://github.com/
- Founded in 2008
- Promotes open source, but also has an Enterprise Edition for businesses

- Offers all Git distributed version control functionality
- Additional functionality:
  - User interface (web-based)
  - Documentation
  - Bug tracking
  - Feature requests
  - Pull requests

GitHub

GitHub basics – forking



GitHub - Original

Version Database

V1

V2

V3

GitHub - Fork

Version Database

V1

V2

V3

upstream

origin

Local Machine

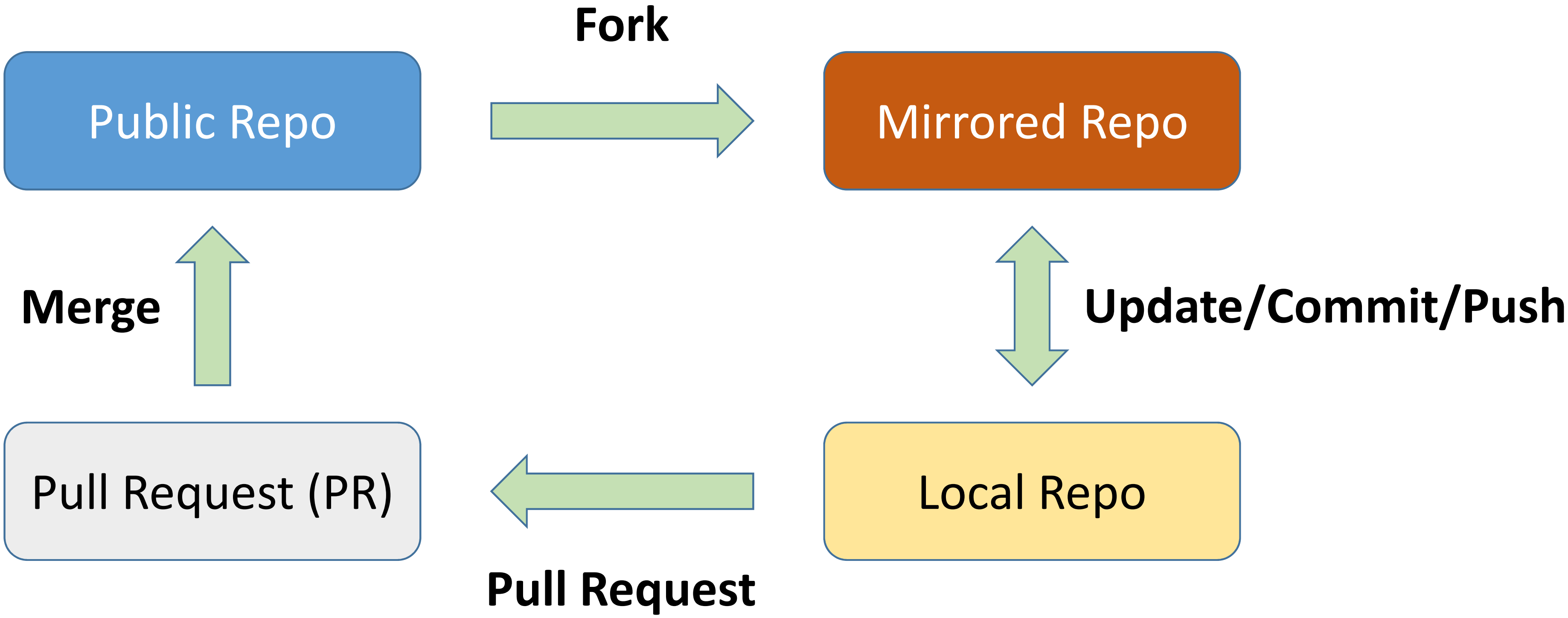
Version Database

V1

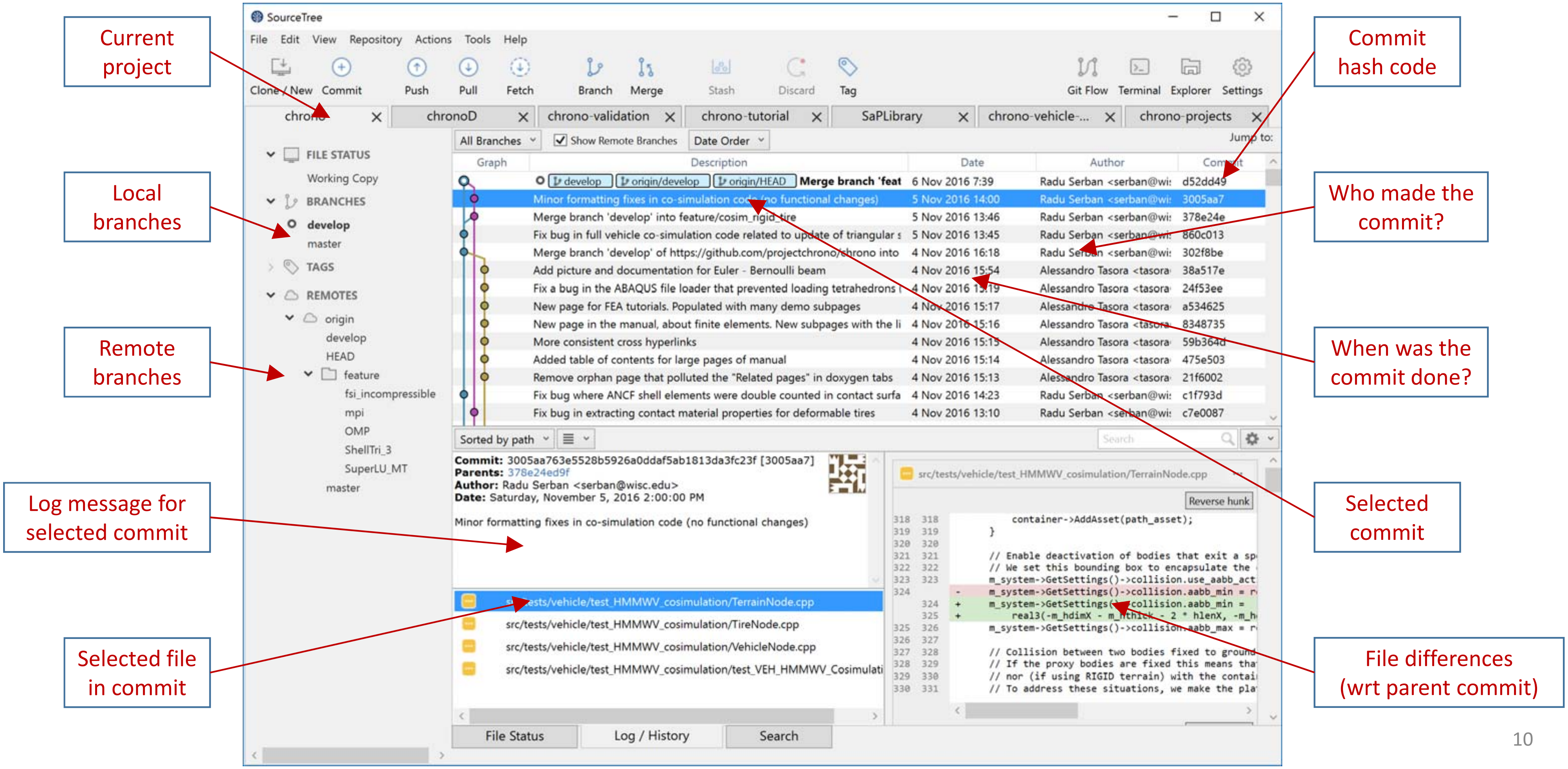
V2

V3

GitHub basics – pull requests



Sourcetree GUI client



Current project

Local branches

Remote branches

Log message for selected commit

Selected file in commit

Commit hash code

Who made the commit?

When was the commit done?

Selected commit

File differences (wrt parent commit)

- FILE STATUS
  - Working Copy
- BRANCHES
  - develop
  - master
- TAGS
- REMOTES
  - origin
    - develop
    - HEAD
    - feature
      - fsi_incompressible
      - mpi
      - OMP
      - ShellTri_3
      - SuperLU_MT
    - master

Graph	Description	Date	Author	Commit
	○ develop origin/develop origin/HEAD **Merge branch 'feat**	6 Nov 2016 7:39	Radu Serban <serban@wi:	d52dd49
	Minor formatting fixes in co-simulation code (no functional changes)	5 Nov 2016 14:00	Radu Serban <serban@wi:	3005aa7
	Merge branch 'develop' into feature/cosim_rigid_tire	5 Nov 2016 13:46	Radu Serban <serban@wi:	378e24e
	Fix bug in full vehicle co-simulation code related to update of triangular s	5 Nov 2016 13:45	Radu Serban <serban@wi:	860c013
	Merge branch 'develop' of https://github.com/projectchrono/chrono into	4 Nov 2016 16:18	Radu Serban <serban@wi:	302f8be
	Add picture and documentation for Euler - Bernoulli beam	4 Nov 2016 15:54	Alessandro Tasora <tasora	38a517e
	Fix a bug in the ABAQUS file loader that prevented loading tetrahedrons (	4 Nov 2016 15:19	Alessandro Tasora <tasora	24f53ee
	New page for FEA tutorials. Populated with many demo subpages	4 Nov 2016 15:17	Alessandro Tasora <tasora	a534625
	New page in the manual, about finite elements. New subpages with the li	4 Nov 2016 15:16	Alessandro Tasora <tasora	8348735
	More consistent cross hyperlinks	4 Nov 2016 15:15	Alessandro Tasora <tasora	59b364d
	Added table of contents for large pages of manual	4 Nov 2016 15:14	Alessandro Tasora <tasora	475e503
	Remove orphan page that polluted the "Related pages" in doxygen tabs	4 Nov 2016 15:13	Alessandro Tasora <tasora	21f6002
	Fix bug where ANCF shell elements were double counted in contact surfa	4 Nov 2016 14:23	Radu Serban <serban@wi:	c1f793d
	Fix bug in extracting contact material properties for deformable tires	4 Nov 2016 13:10	Radu Serban <serban@wi:	c7e0087

Commit: 3005aa763e5528b5926a0ddaf5ab1813da3fc23f [3005aa7]
Parents: 378e24ed9f
Author: Radu Serban <serban@wisc.edu>
Date: Saturday, November 5, 2016 2:00:00 PM

Minor formatting fixes in co-simulation code (no functional changes)

- src/tests/vehicle/test_HMMWV_cosimulation/TerrainNode.cpp
- src/tests/vehicle/test_HMMWV_cosimulation/TireNode.cpp
- src/tests/vehicle/test_HMMWV_cosimulation/VehicleNode.cpp
- src/tests/vehicle/test_HMMWV_cosimulation/test_VEH_HMMWV_Cosimulati

src/tests/vehicle/test_HMMWV_cosimulation/TerrainNode.cpp

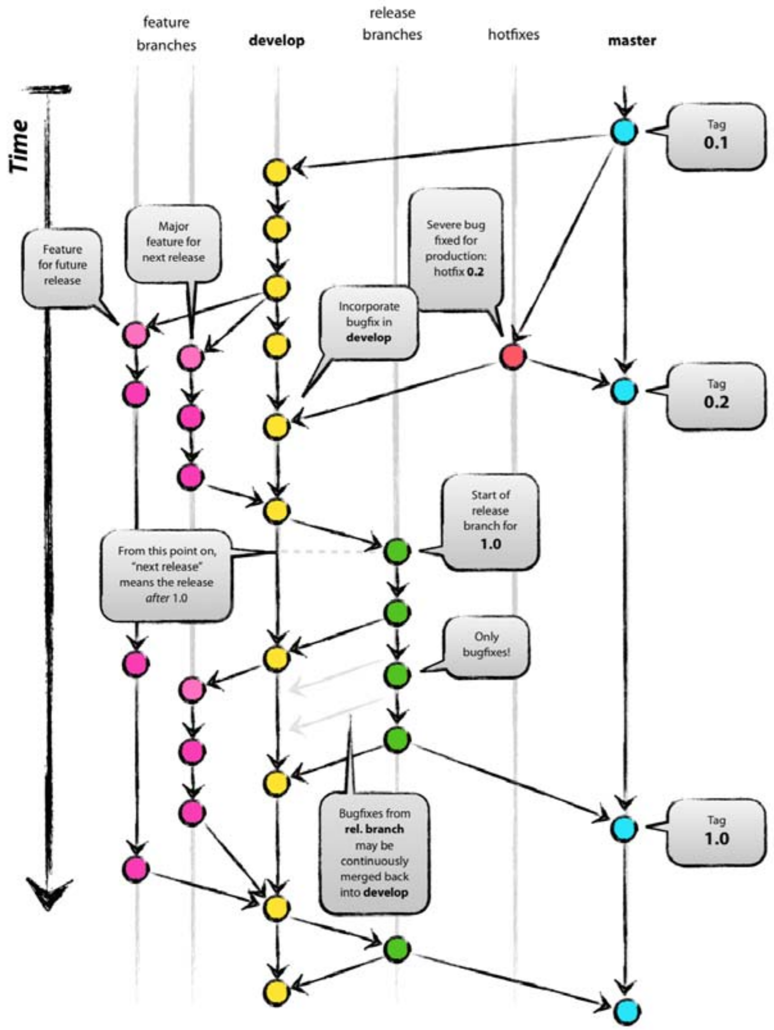
```
        container->AddAsset(path_asset);
    }

    // Enable deactivation of bodies that exit a sp
    // We set this bounding box to encapsulate the
    m_system->GetSettings()->collision.use_aabb_act
-   m_system->GetSettings()->collision.aabb_min = r
+   m_system->GetSettings()->collision.aabb_min =
+       real3(-m_hdimX - m_hthick - 2 * hlenX, -m_h
    m_system->GetSettings()->collision.aabb_max = r

    // Collision between two bodies fixed to ground
    // If the proxy bodies are fixed this means tha
    // nor (if using RIGID terrain) with the contai
    // To address these situations, we make the pla
```

GitFlow

- Proposed by Vicent Driessen
  http://nvie.com/posts/a-successful-git-branching-model/
- A development model based on Git
- Adopted in ProjectChrono


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Tag 0.2
Tag 1.0
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Start of release branch for 1.0
From this point on, "next release" means the release *after* 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop

Git/GitHub resources

- Git official website: https://git-scm.com
- GitHub guides: https://guides.github.com
- Git tutorials and training: https://www.atlassian.com/git/tutorials/
- Git cheat-sheet: https://services.github.com/kit/downloads/github-git-cheat-sheet.pdf
- GetFlow cheat-sheet: http://danielkummer.github.io/git-flow-cheatsheet/

- Git GUI clients: many options (see https://git-scm.com/download/gui/linux)
  - Windows/Mac: Sourcetree (by Atlassian)
  - Linux/Windows/Mac: SmartGit, git-cola