DEPARTMENT OF COMPUTER & INFORMATION SCIENCES

University of Strathclyde Science

Software Engineering

Lecture 3: Configuration Management

Billy Wallace
w.wallace@strath.ac.uk

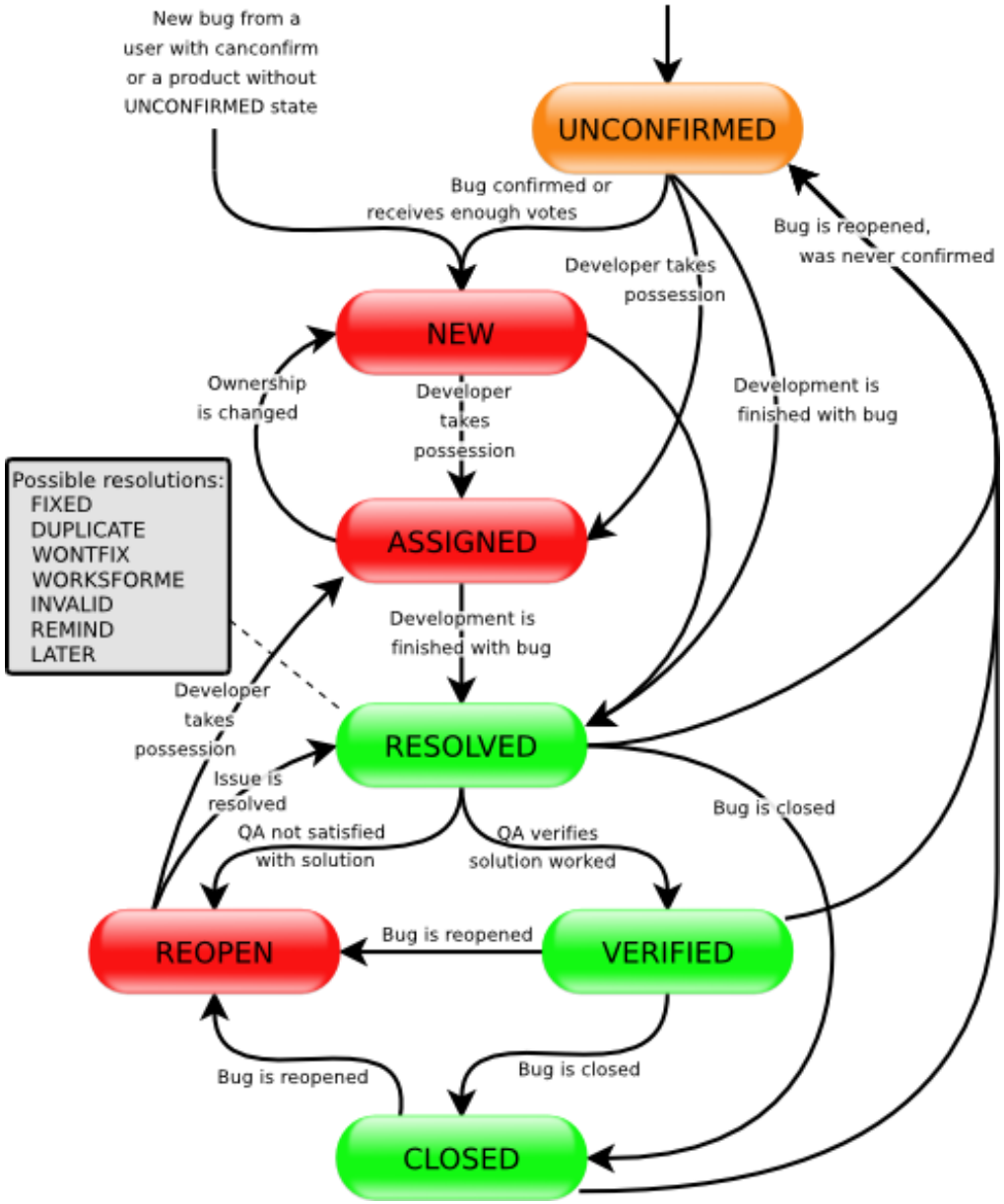
CS993

Lecture Outline

- Configuration management.
- Version Control.
- Other build tools.

What is configuration management?

- Manage the whole process of putting the pieces together.
- Libraries, source, docs, configuration, data.
- This includes release management.


New bug from a user with canconfirm or a product without UNCONFIRMED state
UNCONFIRMED
Bug confirmed or receives enough votes
Bug is reopened, was never confirmed
Developer takes possession
NEW
Ownership is changed
Developer takes possession
Development is finished with bug
Possible resolutions:
FIXED
DUPLICATE
WONTFIX
WORKSFORME
INVALID
REMIND
LATER
ASSIGNED
Development is finished with bug
Developer takes possession
RESOLVED
Issue is resolved
Bug is closed
QA not satisfied with solution
QA verifies solution worked
REOPEN
Bug is reopened
VERIFIED
Bug is reopened
Bug is closed
CLOSED

University of Strathclyde Science

Issue Tracking

- Requesting, evaluating and approving software changes.
- In formal settings we might have a Change Control Board (CCB) that approves changes.

Release Management

- Upgrades, installer, instructions, etc.
- Remember we will need different environments to support this: dev, test, stage and live.
- What if we have multiple customers? We can have each customer on a different branch in version control.

University of Strathclyde Science

Compilers

- Takes source and creates object files (or class files for Java).
- Can be configured to give debug info or to optimise the code.
- Generated code can be different for different compilers, versions, options, so must be managed.
- IDE will support different environments.

Build Tools

- Make is the classic build tool. Rules allow dependencies to be created to say that class files can be created from java files by calling javac.
- If we already have the class files, we don't need to rebuild it, so we save time by only building what's required.
- Ant is a better build tool for Java. The compiler is built in, so we don't launch the same process hundreds of times.
- Ivy can add dependency management, but we've been discussing Maven.

Maven

- Maven is ant plus much more.
- It uses conventions to describe the layout of the project. Where the source lives, where the binaries go, how tests work, etc.
- Perhaps more importantly it allows dependencies to be controlled.

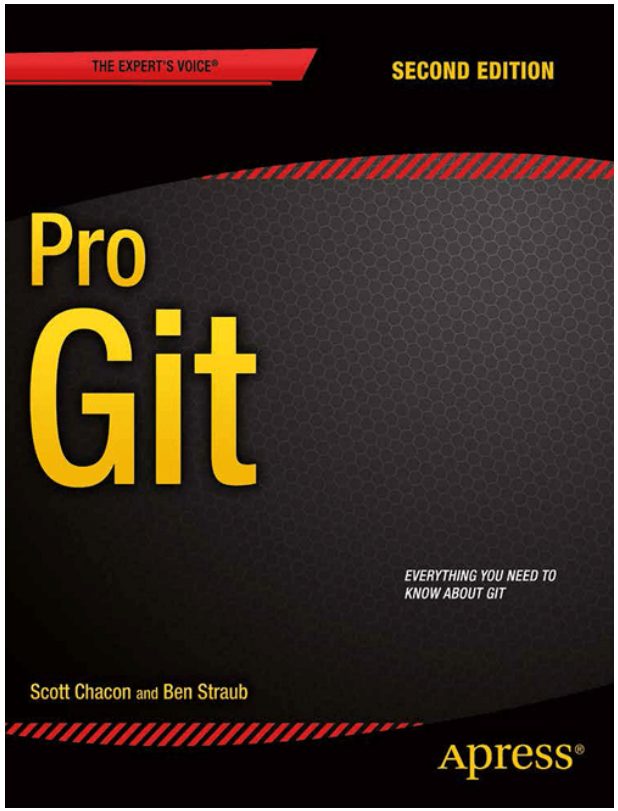
University of Strathclyde Science

Version Control

- SCCS, RCS, CVS, Perforce, Subversion, Mercurial, etc.
- Check-out is old school.
- Now we merge.
- Git is the most popular tool, so worth knowing.

University of Strathclyde Science

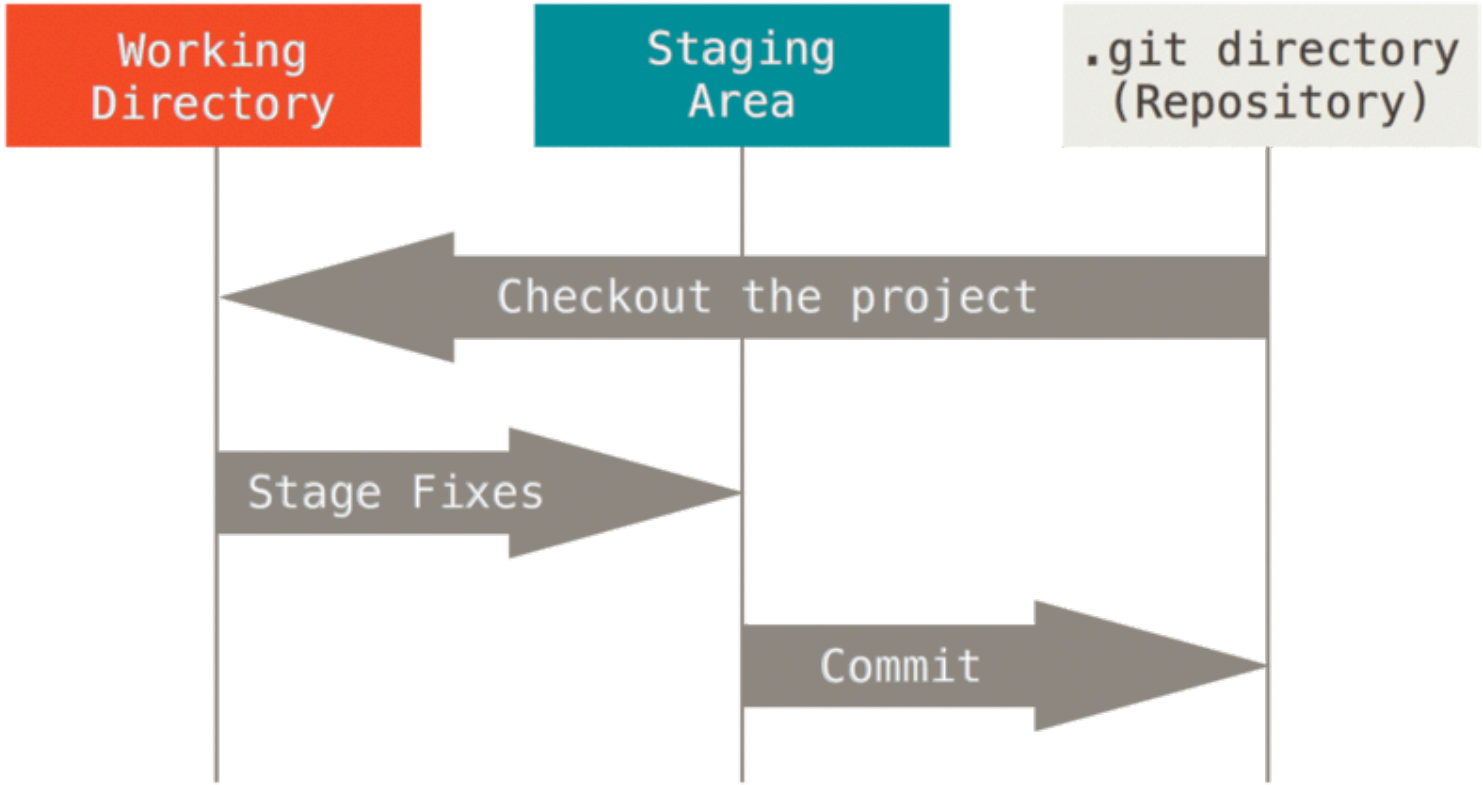
Free Book!



https://git-scm.com/book/en/v2

University of Strathclyde Science

Git Workflow



Git Operations

- Create (init) or clone repository.
- Add and commit.
- Push and pull - remotes.
- Branches and merges.
- Tags.

Summary

- The software we release needs careful configuration. Different versions will go into different environments and potentially to different customers.
- Version control is a central aspect to this.
- We also involve our bug tracker and build tools in this process.

Moving on from here …

- In the practical this week, we'll look at Git.
- Go back and look at the Git operations we did for the first lab when we worked with Heroku.
- Look at the early parts of the Git Basics chapter of the free book. Get yourself comfortable with add, commit and status.

University of Strathclyde Glasgow