# Continuous Integration

Tobias Nyholm





#### About me

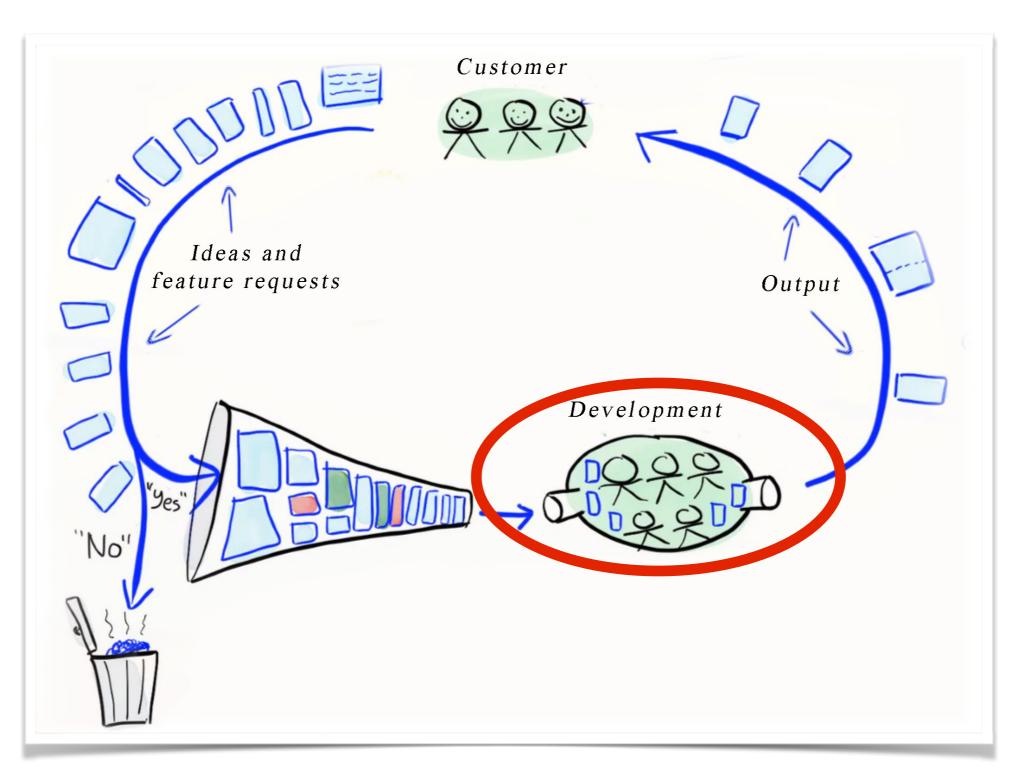
- Doing Symfony since 1.2
- Master thesis at KTH

## • HAPPY RECRUITING

We sort your pile of resumés



### Every software project





### What is efficiency?

- Resource efficiency
- Every developer works full time
- You develop all day without distractions like Facebook or coffee breaks



### What is efficiency?

- Flow efficiency
- Developer A completes X in 4 h
- Developer B completes the same task X in 3 h
- Developer B is more efficient than A



### What is quality?

- Reliability
- Performance Efficiency
- Security
- Maintainability
- Size of the project

Ref: CISQ's Quality model

### Extreme Programming - Definition

- Pair programming
- Plan sprints
- Test Driven Development
- Communication with the customer
- Continuous Integration
- Refactoring
- Small releases
- Coding standards

- Collective code ownership
- Simple design
- System metaphor
- Sustainable pace
- No overtime
- Leave optimization until last
- 100% test coverage
- Write test for all bugs

Ref: http://en.wikipedia.org/wiki/Extreme\_programming

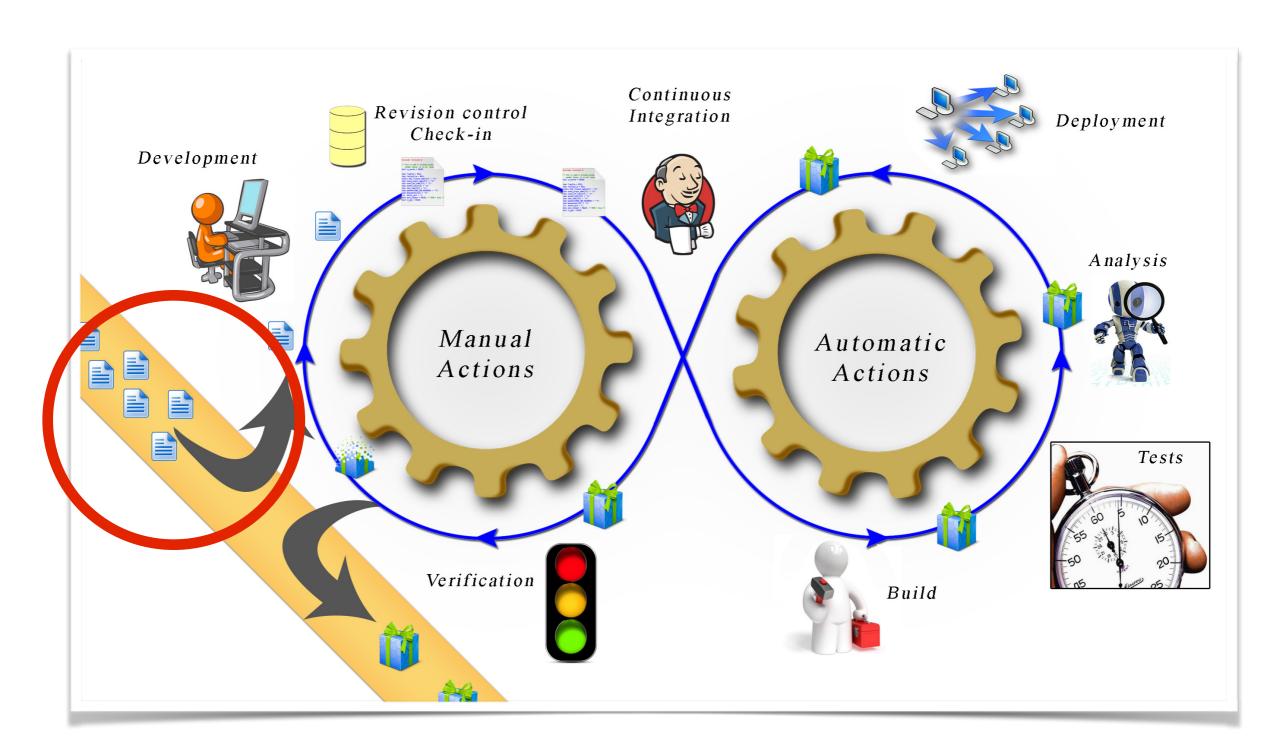


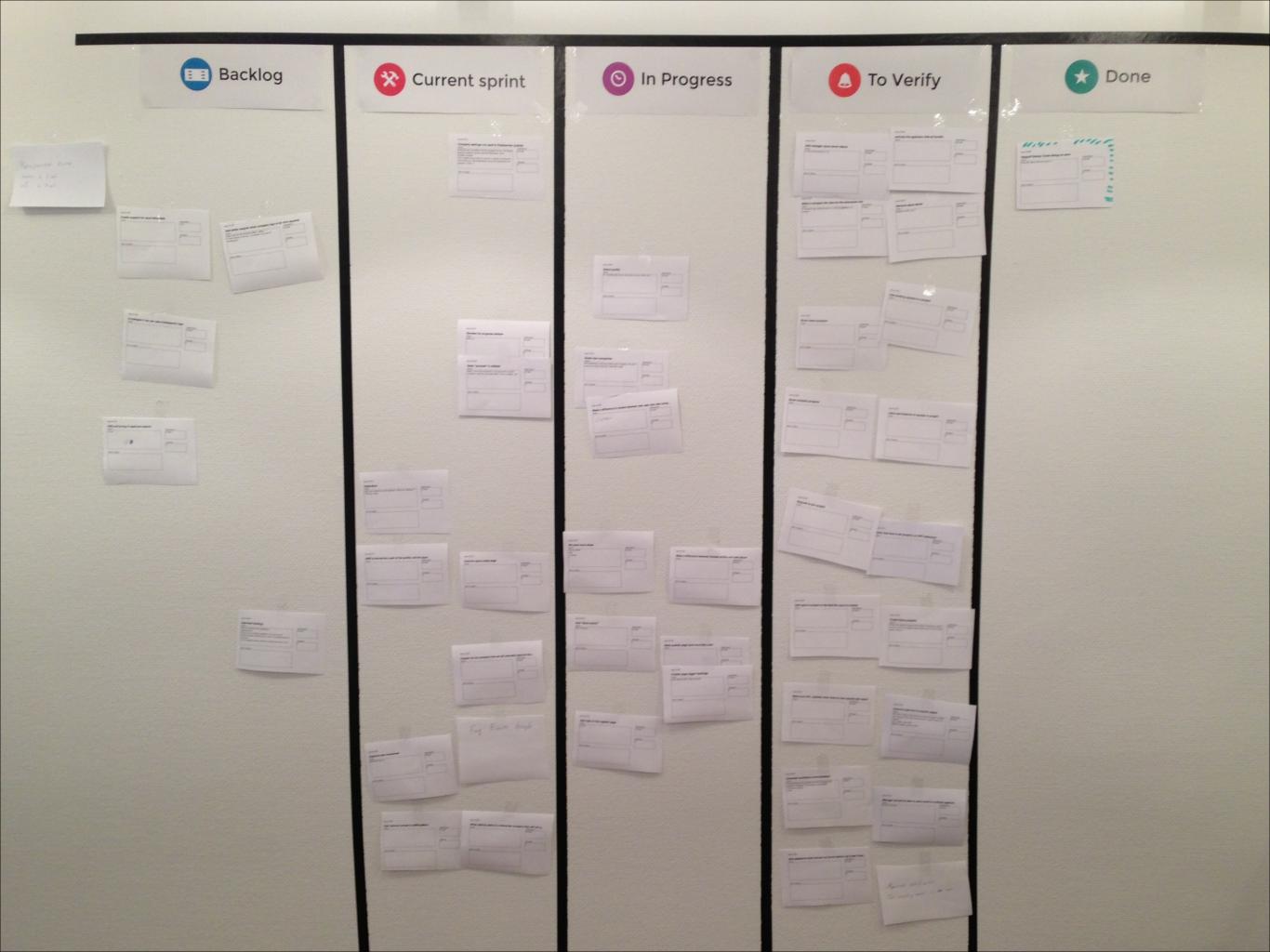
### Extreme Programming

- KISS and DRY
- Test Driven Developmet
- Trust your test code
- Share knowledge between developers
- Be happy and healthy



### Overview





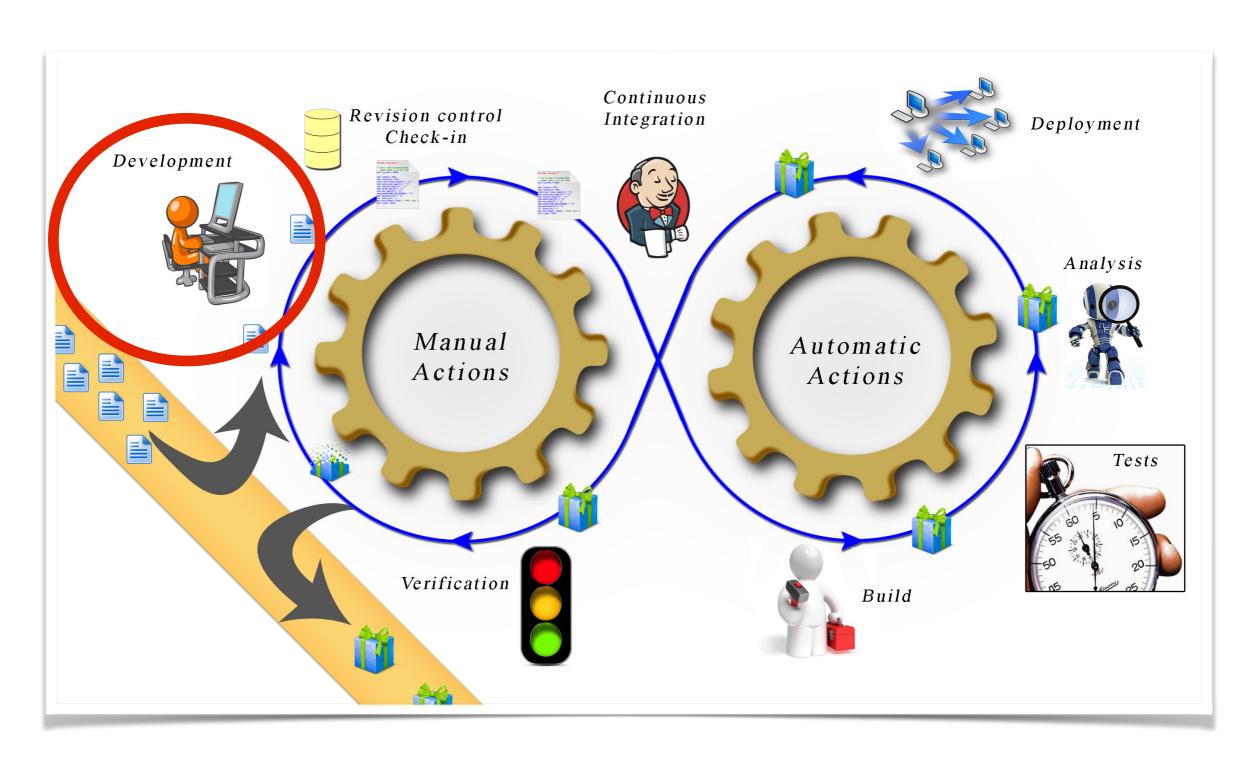


### Story Cards

See your o	wn transaction history	3
Notes Use paging to avoid	large DB queries. Design similar to view users page.	Estimate
How to demo Log in, click on "trans that the new deposit	sactions". Do a deposit. Go back to transactions, check shows up.	

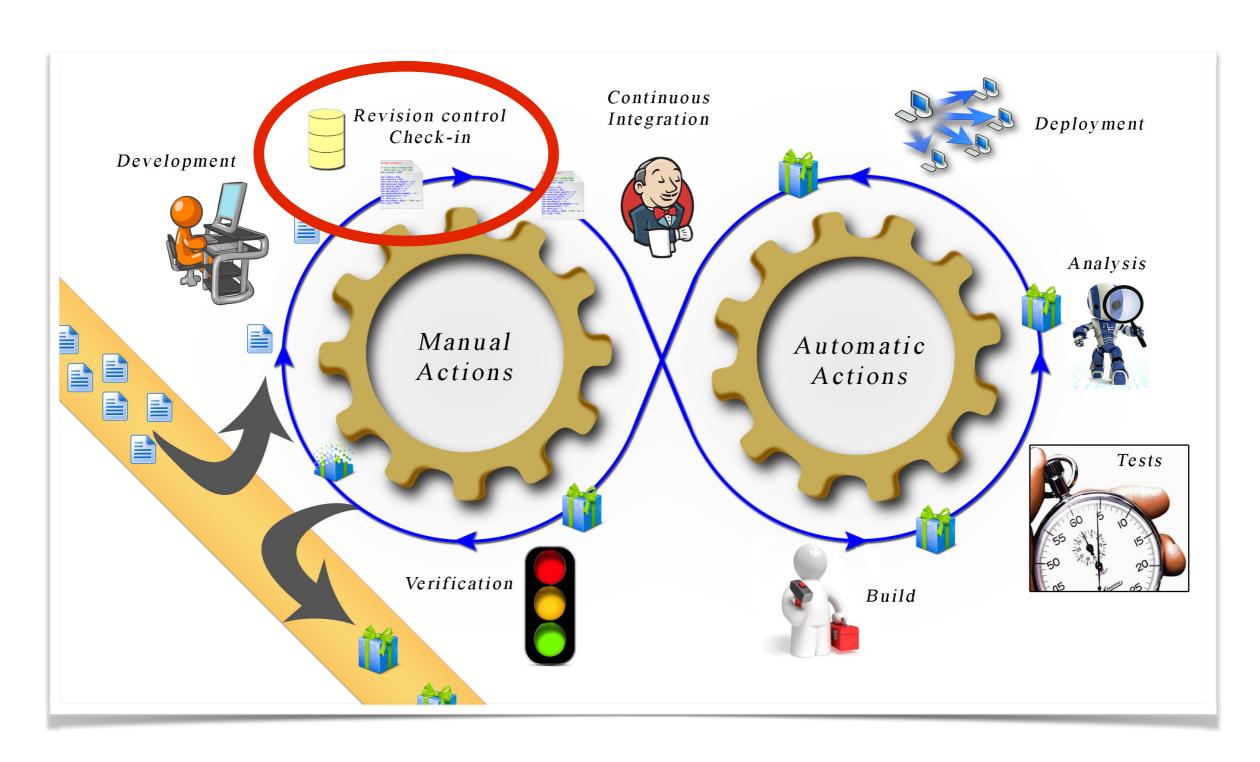


### Overview - Development



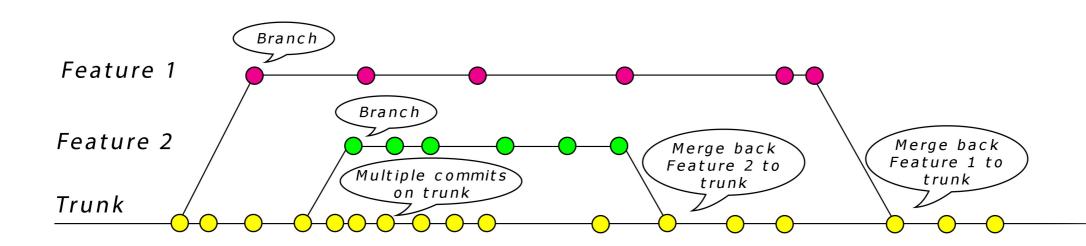


#### Overview - Revision control



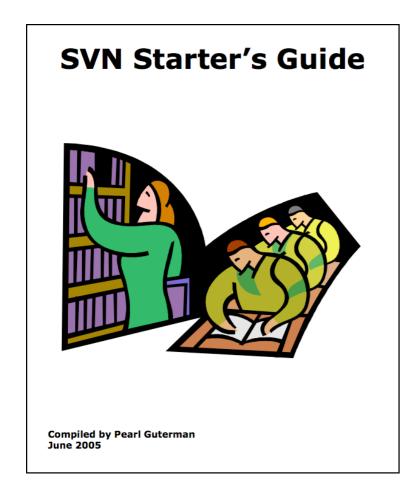


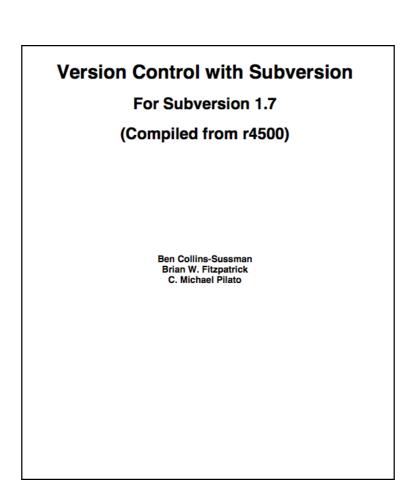
### Branching

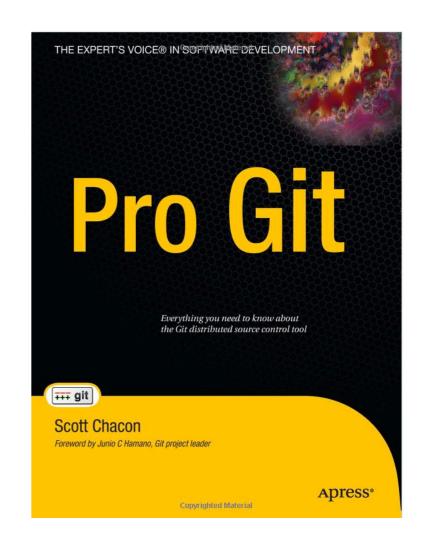


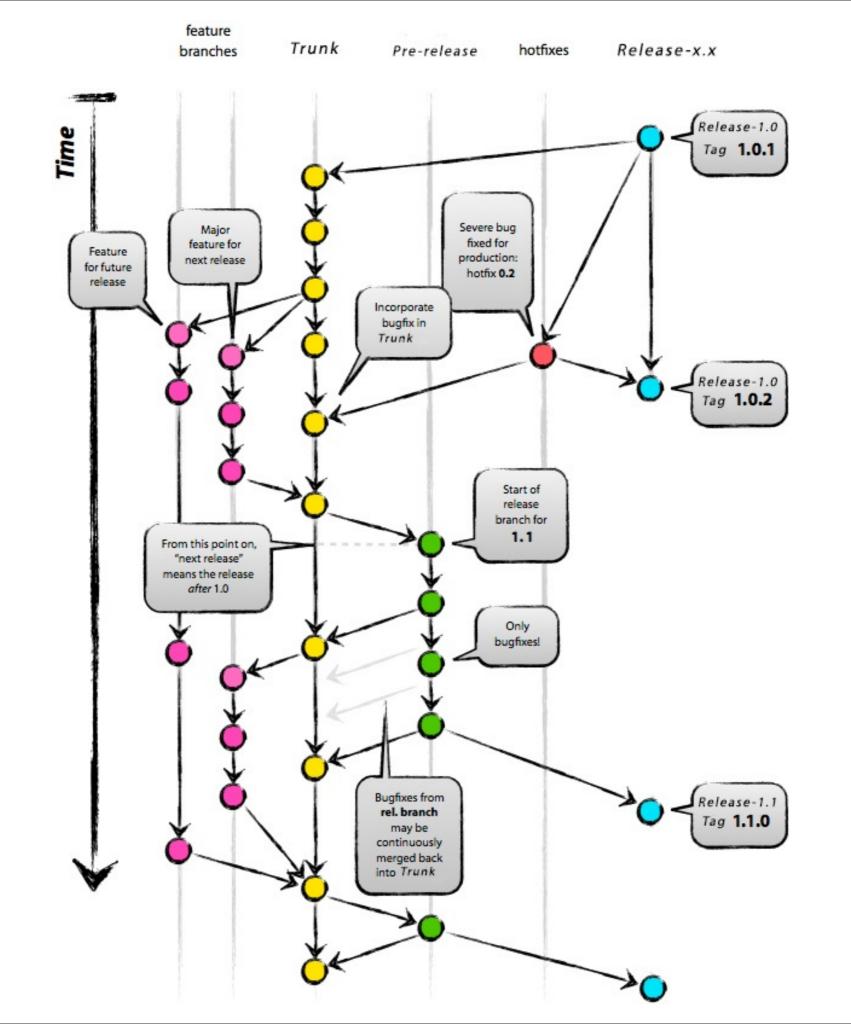


#### Git vs SVN





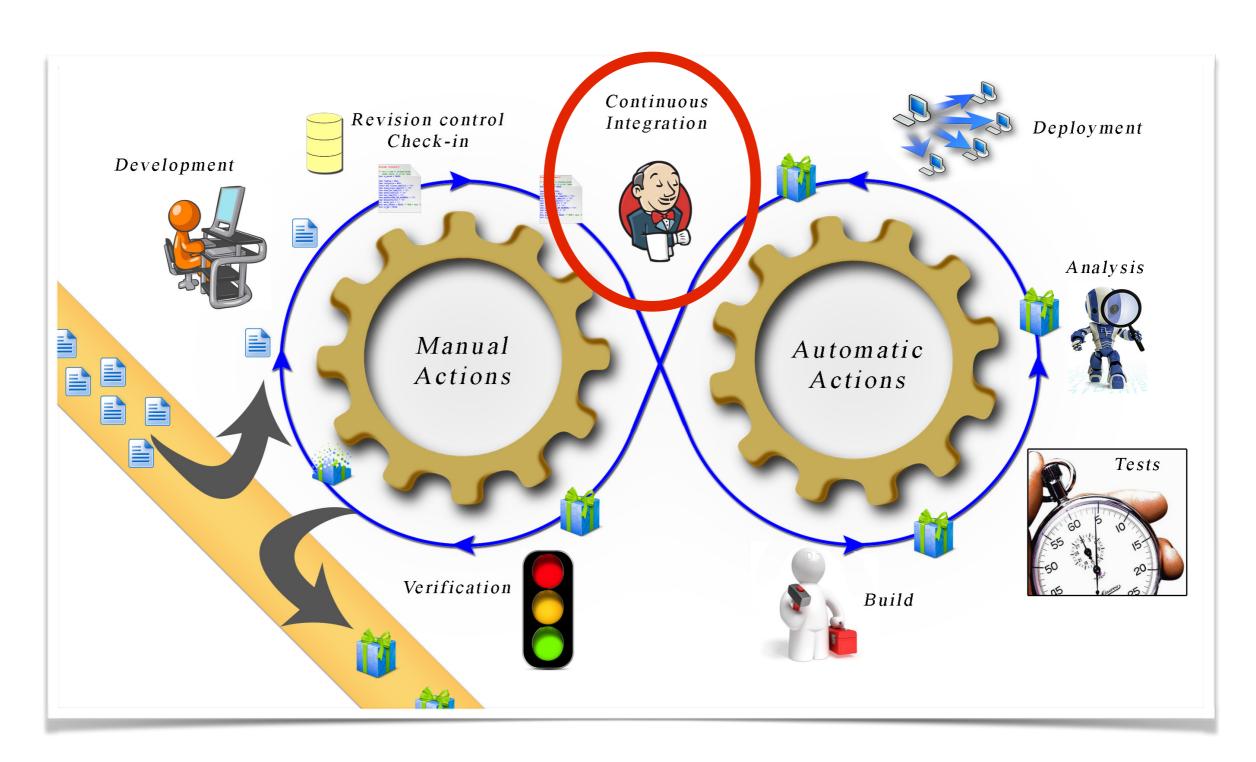






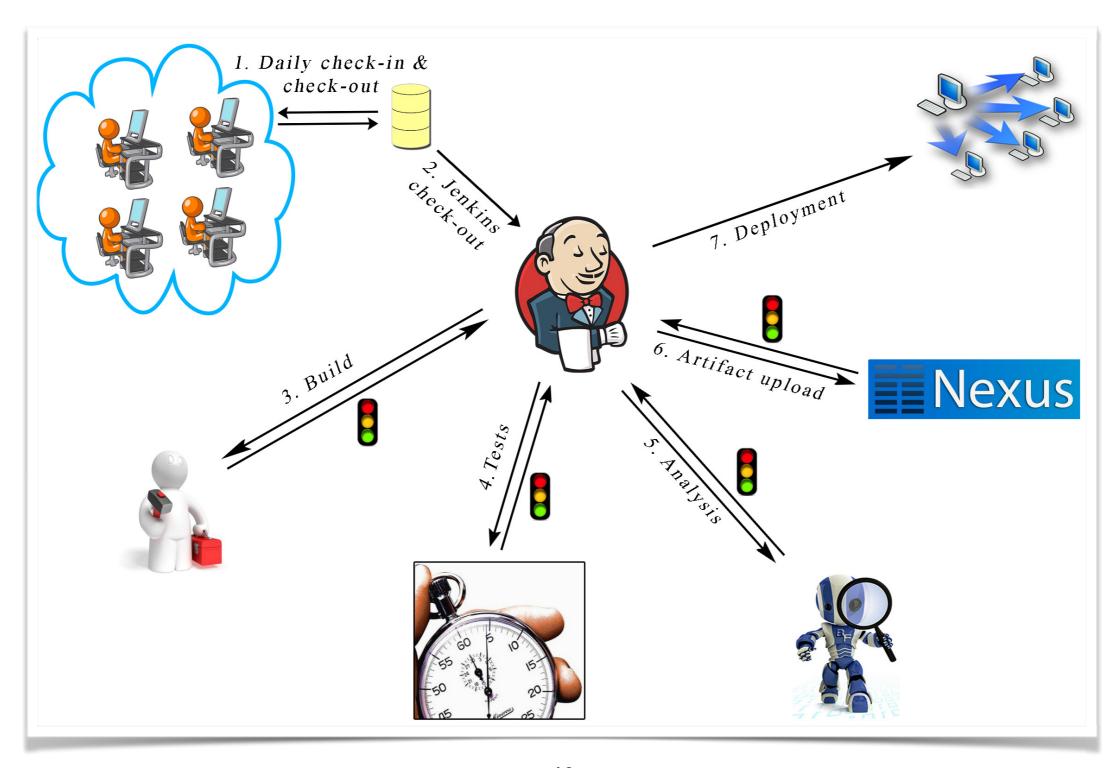


### Overview - Jenkins





### Jenkins is the heart of CI



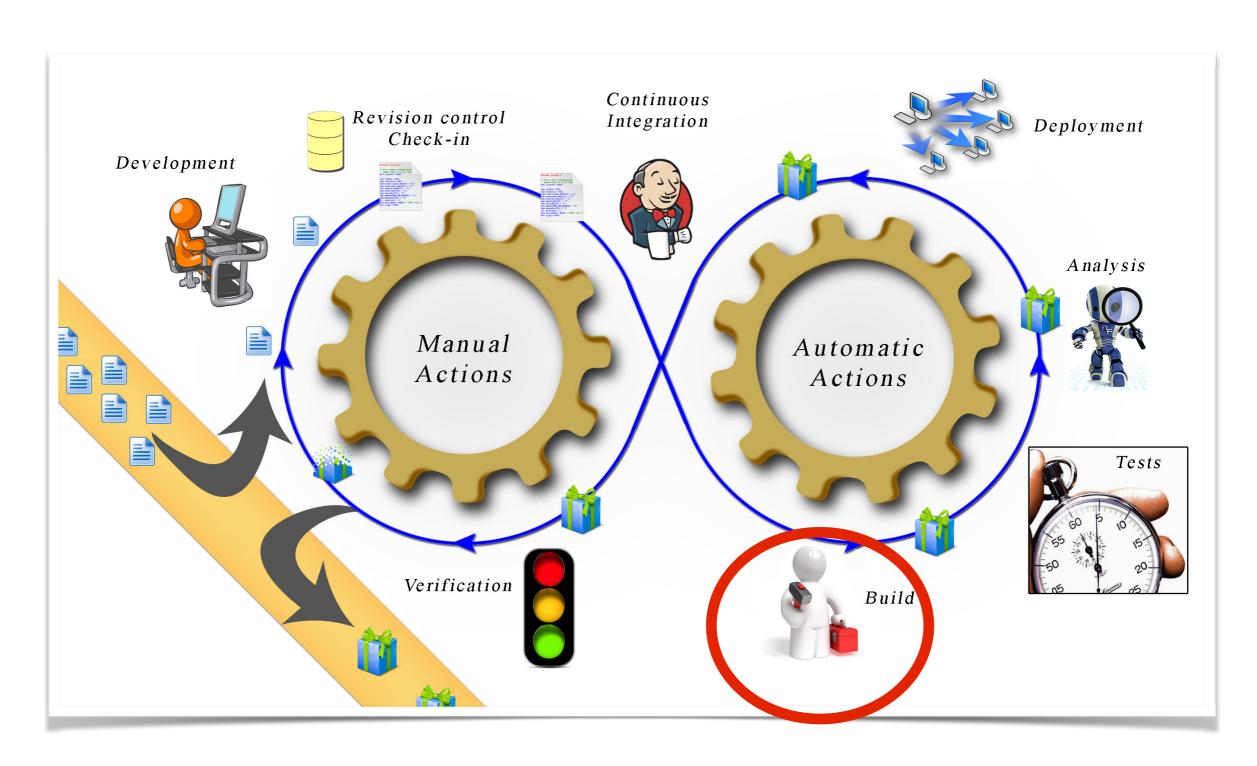


### Important stuff

- Quick feedback
- SVN/GIT hook
- Only CM should have permissions to change the rules in Jenkins



#### Overview - Build servers



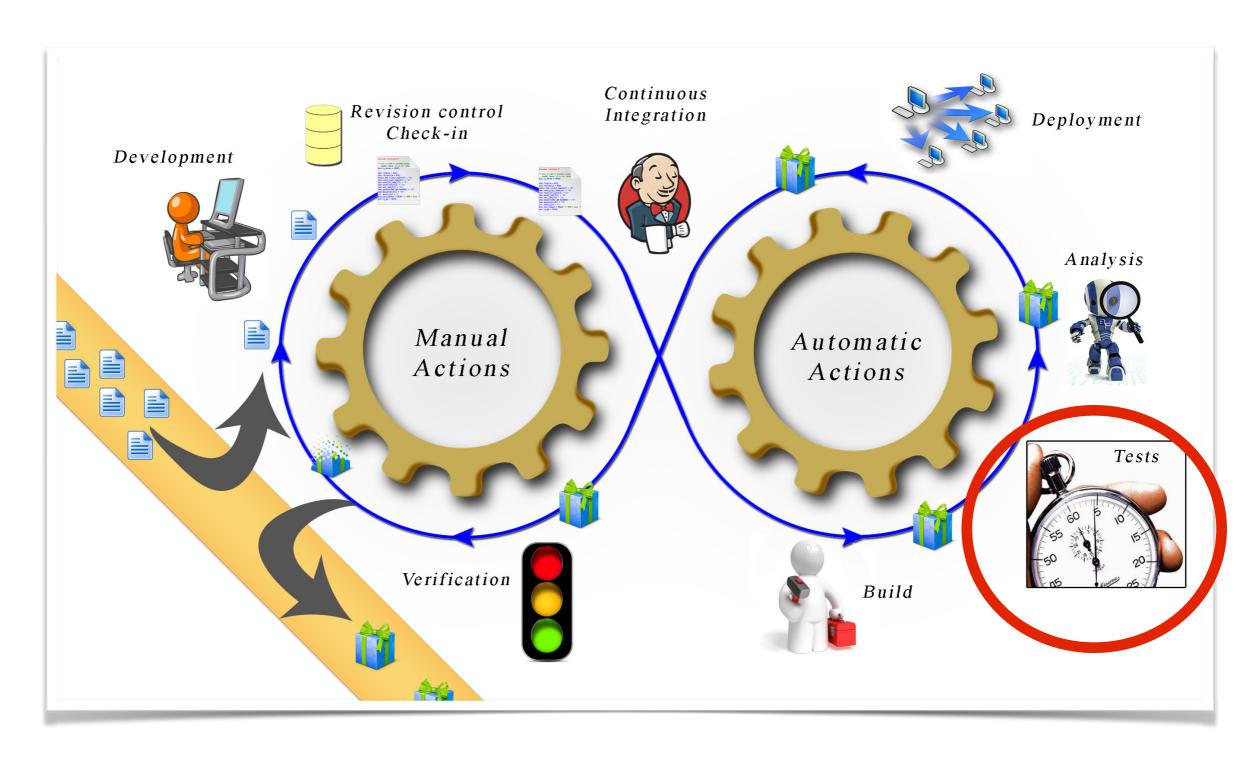


#### Build fast

- Build SNAPSHOTS from master
- Short build queues (< 10 min)</li>
  - Multiple build servers
  - Parallel builds

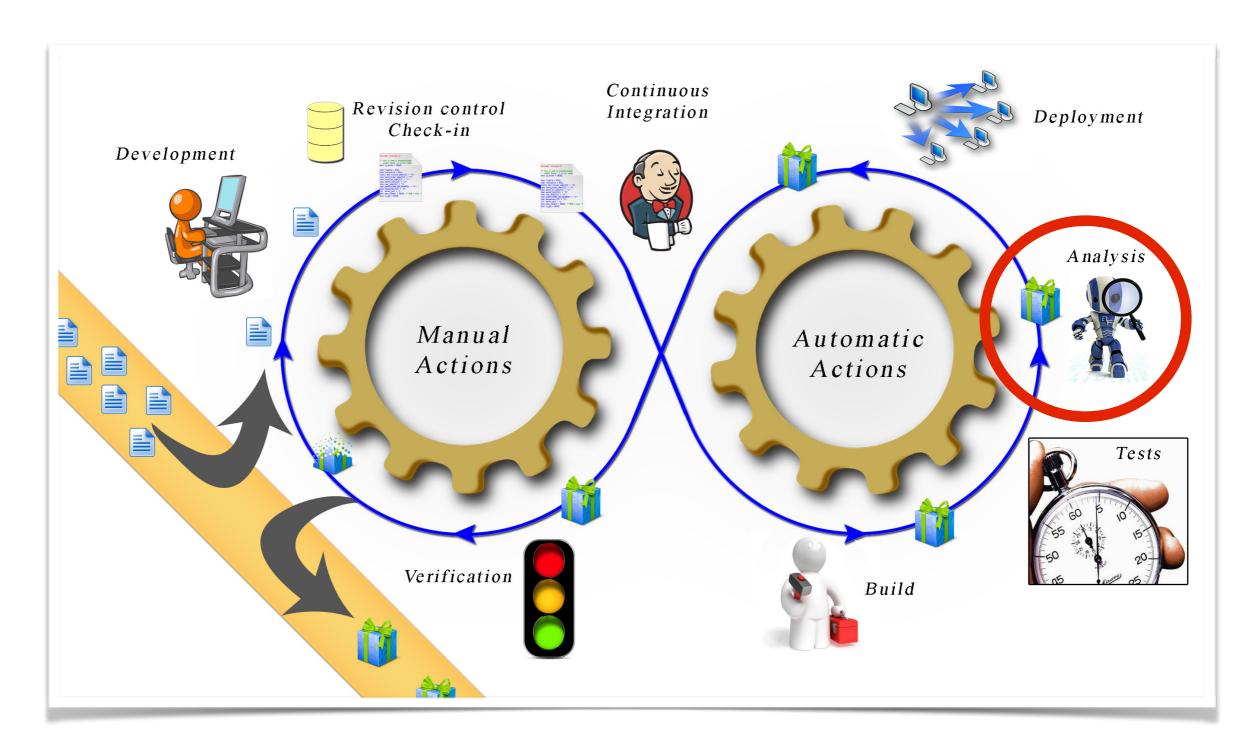


#### Overview - Test





### Overview - Analysis





#### Sonar

- A platform for code analysis
- Make you own quality profile
- Create alarms when the profile is violated
- Break the build when an alarm triggered



#### Sonar - Rules

- Plugins like PHPMD, PHP Depend, PHP\_CodeSniffer etc
  - Complexity
  - Comments
  - Size of the source code
  - Performance



#### Sonar - Benefits

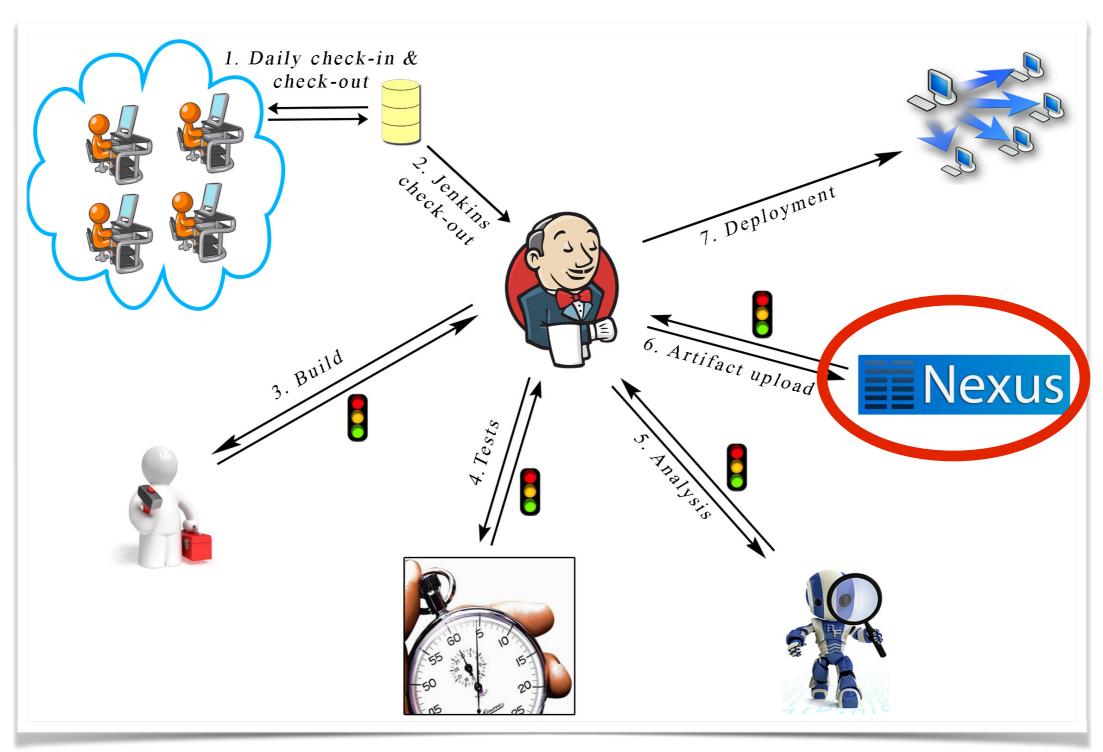
- Change history
- Unified code that is easy to maintain
- Reduce complexity
- More documented code

#### **Quality - Definition**

- **√**Reliability
- ✓ Performance Security
- **√** Maintainability
- √Size of the project

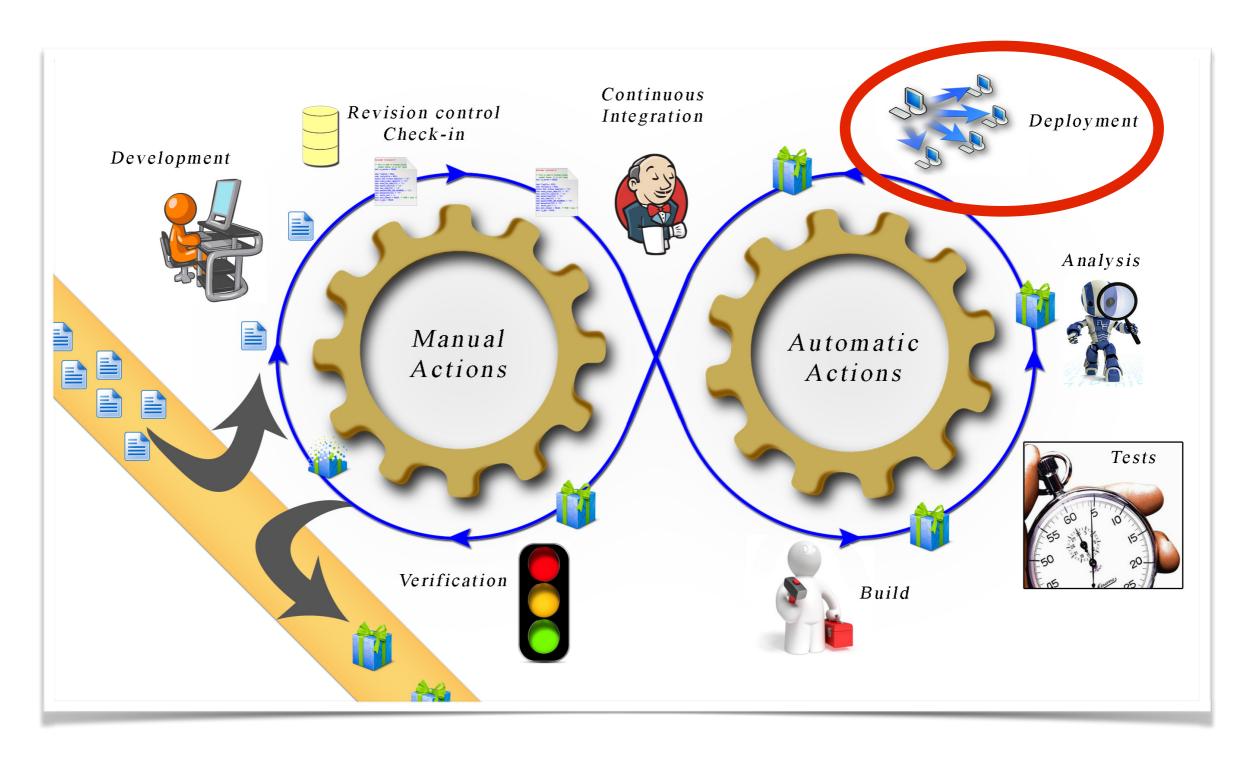


### Upload to artefact repository





### Overview - Deployment





### Puppet is king





### Capistrano, Ant, Octopus

- Check out code
- Migrate the database
- Empty cache, warm up cache
- Run scripts
- etc



#### Distribution

- Distribute file automatically att each build
- No developer should manually update any server
- No human errors

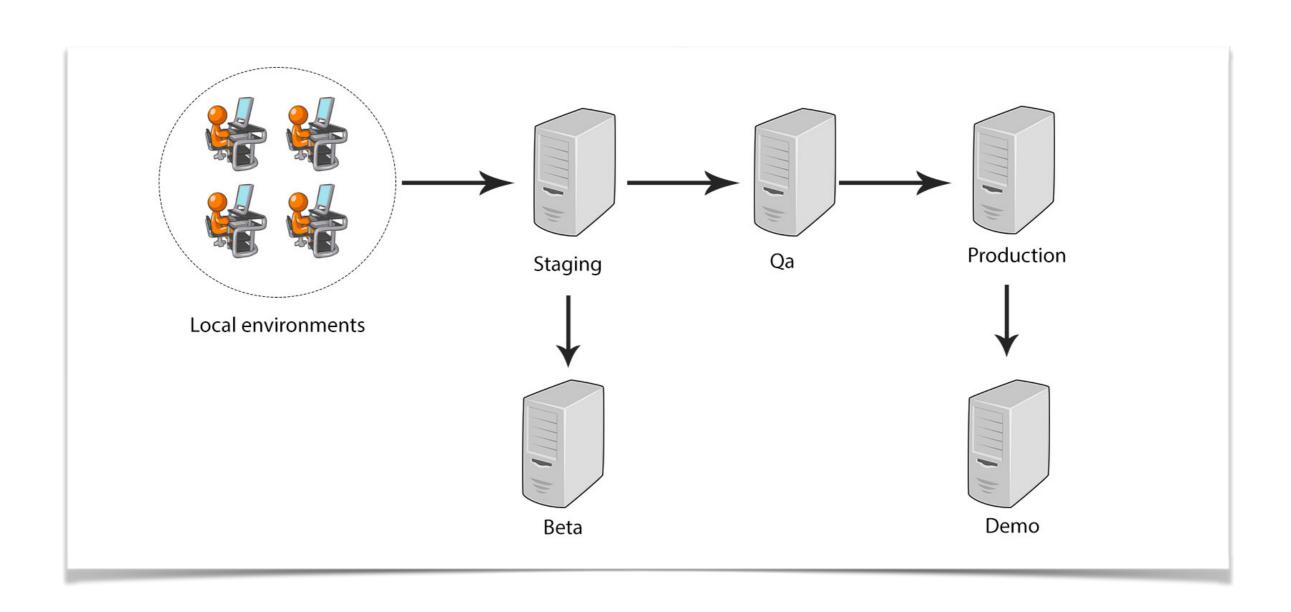


#### Environments

- Local environment of every developers machine
- Staging environment where Jenkins moves files automatically
- QA environment where you move files when needed
- Production environment
- Make sure that the customer or project manager provide you with these

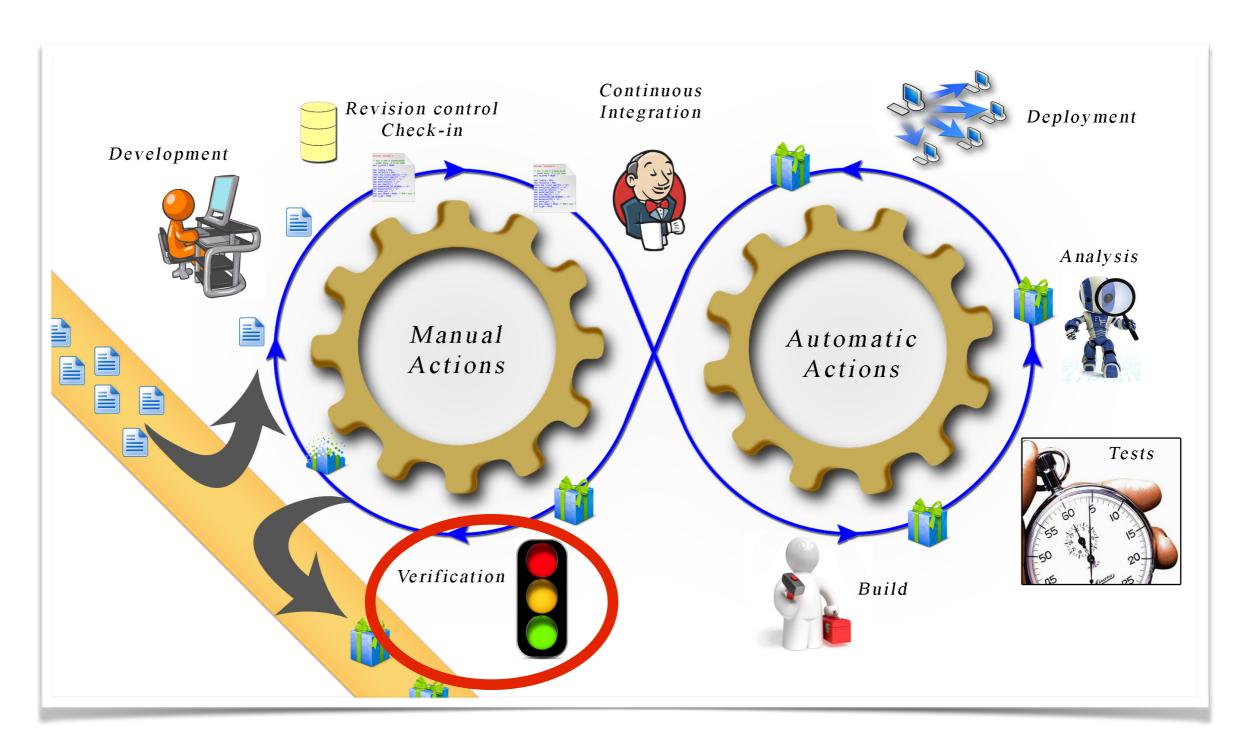


### On HappyRecruiting



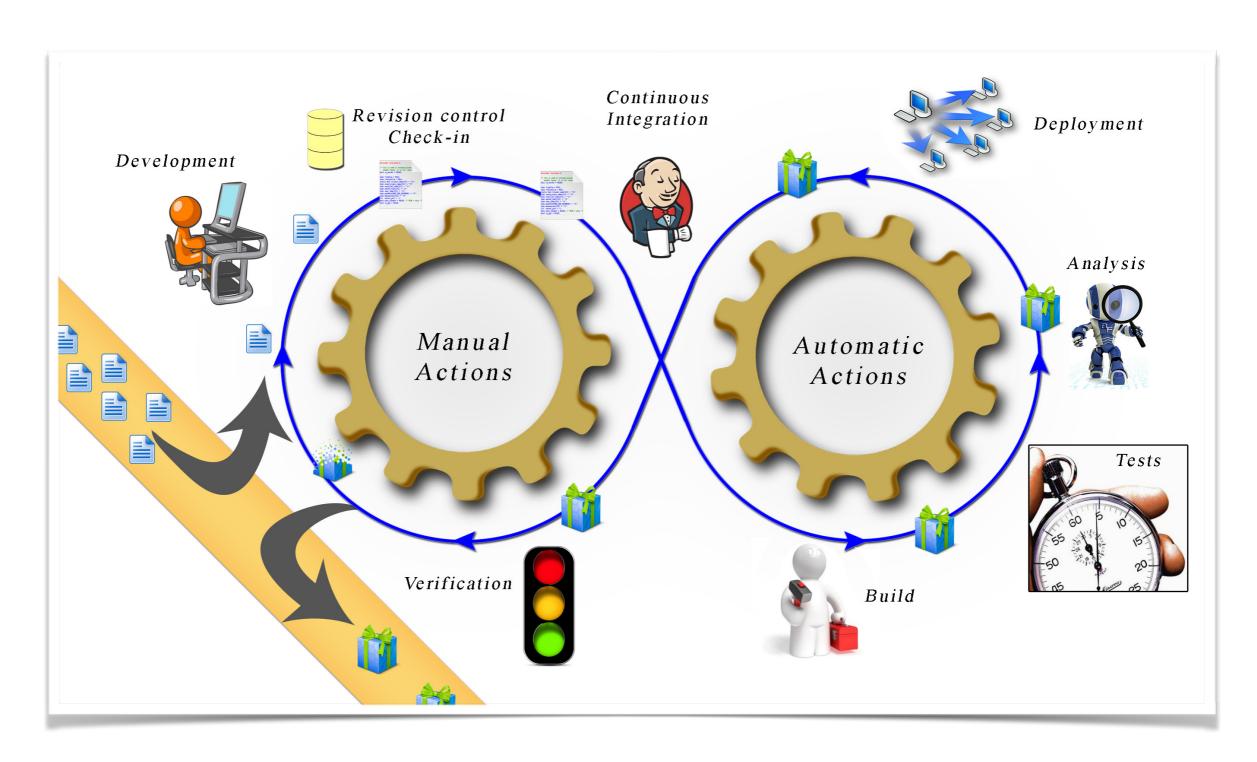


#### Overview - Verification





### Summary





### Thank you

### Tobias Nyholm

2013-11-22