## **Programutveckling med Git**

#### En introduktion

Most images in this presentation are from the Pro Git book. The entire <u>Pro Git book (https://git-scm.com/book/en/v2)</u>, written by Scott Chacon and Ben Straub and published by Apress, is available here. All content is licensed under the <u>Creative Commons Attribution Non Commercial Share Alike 3.0 license</u> (<u>https://creativecommons.org/licenses/by-nc-sa/3.0/</u>)</u>. Print versions of the book are available on Amazon.com.

Some images are from <u>Git With The Program: Introduction to Basic Git Concepts - Drupal Camp LA 2013</u> (<u>https://www.slideshare.net/mwrather/git-with-the-program-dcla-2013</u>). All content is licensed under the <u>CC</u> <u>Attribution-ShareAlike License (https://creativecommons.org/licenses/by-sa/4.0/)</u>.

### Why Git

- Share information with others and/or yourself
- Work with multiple things at the same time
- Backup
- Test stuff quick and easy
- So many projects are using Git

#### In [1]:

```
from IPython.core.display import HTML
HTML('<iframe width="560" height="315" src="https://www.youtube.com/embed/pOSqctH</pre>
```

Out[1]:



#### **Version Control Systems**

#### To mention a few:

- Centralized Version Control Systems
  - Git
  - Mercurial
- Distributed Version Control Systems
  - SCCS
  - RCS
  - CVS
  - PVCS
  - SourceTree
  - TFS
  - Perforce



### **Git concepts**

### Stream of snapshots



### List of file-based changes



### Data Model

- Blobs (atual data)
- Tree (directories of blobs or of other trees)
- A commit
  - A tree
  - Zero or more parent commits
  - Meta data (message, email, timestamp)



SHA1

42a75c677b6c87aede617e79ff912a617e3ccb3118 fec67c95ec9f32136d9a1a90a750d4005f46f36746 b9a80d63efb64a1def1125eb5eaece68b649a7f2a1 4ca6eb6f65eed45ec8f6c87cfe1b9c9cdea095e774 96d5e67d606c8681aff2341c47298ce4957d8ab8e8 c**6bcb34450fc101fbf885ea7a320a425f6d2475d3**a 0afa7178b8e78fdace33fd72ed4f1fdc0128da975a d34991122c3c23ac13b1496232ef3b2d286a58fce6 3e8be37cf6c771174195e0c5acb27b4085f4906dfb 67b8a100276bc8068ce7c4fedba273bba980c14f51 8e10af0ee3e36116084a394e285f1b023c95ba324f 70b0cd84b08d20c48fadbc5f3743ba3459aeb351ff 13f51f2fdbaf93c59c6e9434528b1fd62207f331b5

**The Three States** 



### Let's get started

- Install Git. Start here: <u>https://git-scm.com (https://git-scm.com)</u>
- First time setup on local machine
  - git config --global user.name "John Doe"
  - git config --global user.email johndoe@example.com
  - git config --global core.editor "'C:/Program Files (x86)/Notepad++/notepad++.exe'
- Remote git server (optional)
  - <u>https://github.com/join?source=header-home (https://github.com/join?source=header-home)</u>
  - https://bitbucket.org/account/signup/ (https://bitbucket.org/account/signup/)
  - <u>https://gitlab.com/users/sign\_in (https://gitlab.com/users/sign\_in)</u>

### Initialize a local repository

In [2]:

```
%%bash
cd /tmp
rm -rf TickProject
mkdir TickProject
cd TickProject
git init
tree -aC
```

Initialized empty Git repository in /private/tmp/TickProject/.git/

```
- .git
  ├── HEAD
  ├-- config
  --- description
   - hooks
      --- applypatch-msg.sample
      ├── commit_msg.sample
      ├-- post-update.sample
      ├── pre-applypatch.sample
      ├── pre-commit.sample
      --- pre-push.sample
      ├── pre-rebase.sample
      ├── pre-receive.sample
      --- prepare-commit-msg.sample
      └── update.sample
    - info
      └── exclude
    - objects
      ├── info
      └── pack
    - refs
      ├── heads
      └── tags
```

```
9 directories, 14 files
```

### Write some code and check status

In [3]:

```
nothing added to commit but untracked files present (use "git add" t o track)
```

### Add code to stage area and check status

### Transfer code to local repository

```
In [5]:
```

```
%%bash
cd /tmp/TickProject
git commit -a -m "Initial revision"
git status
[master (root-commit) 19b738a] Initial revision
3 files changed, 109 insertions(+)
create mode 100644 src/KeyTicker.h
create mode 100644 src/TickerInterface.h
create mode 100644 src/main.cpp
On branch master
nothing to commit, working tree clean
```

### **Some Git clients**

- SourceTree <u>https://www.sourcetreeapp.com (https://www.sourcetreeapp.com)</u>
- GitKraken <u>https://www.gitkraken.com (https://www.gitkraken.com)</u>
- TortoiseGit <u>https://tortoisegit.org (https://tortoisegit.org)</u>
- Eclipse plugins
- XCode builtin

# Write some more code (change name of a variable and add a meson.build file)

Use a gui client to see status and commit code

```
In [6]:
```

```
!. quickedit v0.2
```

### **Build and check status**

- <u>Meson (http://mesonbuild.com)</u> is a build generating system
- <u>Ninja (https://ninja-build.org)</u> is a small build system with a focus on speed.

In [7]:

```
%%bash
cd /tmp/TickProject && rm -rf build
meson src build
```

```
The Meson build system
Version: 0.39.1
Source dir: /private/tmp/TickProject/src
Build dir: /private/tmp/TickProject/build
Build type: native build
Project name: TickCounterProject
Native cpp compiler: c++ (clang 8.1.0)
Build machine cpu family: x86_64
Build machine cpu: x86_64
Dependency threads found: YES
Build targets in project: 1
```

In [8]:

%%bash
ninja -C /tmp/TickProject/build

```
ninja: Entering directory `/tmp/TickProject/build'
[1/2] Compiling cpp object 'tickcounter@exe/main.cpp.o'
[2/2] Linking target tickcounter
clang: warning: argument unused during compilation: '-pthread' [-Wun
used-command-line-argument]
```

In [9]:

%%bash
cd /tmp/TickProject
git status

```
On branch master
Untracked files:
   (use "git add <file>..." to include in what will be committed)
```

build/

nothing added to commit but untracked files present (use "git add" t o track)

## Instruct Git to ignore build artifacts and other files that shouldn't be version controlled

Add an .gitignore file There are plenty of examples of .gitignore files on the net and/or use e.g., <u>https://www.gitignore.io (https://www.gitignore.io)</u>

```
In [10]:
```

%%bash cd /tmp/TickProject git status

On branch master Untracked files: (use "git add <file>..." to include in what will be committed)

.gitignore

nothing added to commit but untracked files present (use "git add" t o track)

### **Branches, HEAD and tags?**



In [11]:

%%bash cd /tmp/TickProject git branch -v

\* master 6845752 Initial revision

**Creating a New Branch** 



#### Switching Branch



#### The HEAD branch moves forward when a commit is made



### **Branching models**

**Git-flow** <u>http://nvie.com/posts/a-successful-git-branching-model/ (http://nvie.com/posts/a-successful-git-branching-model/)</u>



**Cactus** <u>https://barro.github.io/2016/02/a-succesful-git-branching-model-considered-harmful/</u> (https://barro.github.io/2016/02/a-succesful-git-branching-model-considered-harmful/)



**BBC News** <u>http://www.integralist.co.uk/posts/github-workflow.html</u> (<u>http://www.integralist.co.uk/posts/github-workflow.html</u>)

**Git Pro book** <u>https://git-scm.com/book/en/v2/Git-Branching-Branching-Workflows (https://git-scm.com/book/en/v2/Git-Branching-Branching-Workflows)</u>

### Third party package usage strategies

- Package Manager
  - Homebrew
  - Nuget
  - apt-get
- Git submodules / subtrees
  - Qt



- In source
- Separate repositories (pkg-config)

### Add better logging support to our TickCounter project

Let's use the 3rd party spdlog package as a separate repository and try it out in a logging feature branch

https://github.com/gabime/spdlog.git (https://github.com/gabime/spdlog.git)

### Get the spdlog repo

In [12]:

%%bash
cd /tmp
rm -rf spdlog
rm -rf 3rd/spdlog
git clone https://github.com/gabime/spdlog.git spdlog

Cloning into 'spdlog'...

```
In [13]:
%%bash
cd /tmp/spdlog
ls –la
total 32
drwxr-xr-x 15 tommy
                    wheel
                            510 Apr 30 19:33 .
drwxrwxrwt 14 root
                    wheel
                            476 Apr 30 19:33 ..
                    wheel
                            408 Apr 30 19:33 .git
drwxr-xr-x 12 tommy
-rw-r--r-- 1 tommy wheel
                            779 Apr 30 19:33 .gitignore
                           3518 Apr 30 19:33 .travis.yml
-rw-r--r-- 1 tommy wheel
                           2418 Apr 30 19:33 CMakeLists.txt
-rw-r--r-- 1 tommy
                    wheel
                            323 Apr 30 19:33 INSTALL
-rw-r--r-- 1 tommy wheel
-rw-r--r-- 1 tommy wheel 1142 Apr 30 19:33 LICENSE
-rw-r--r-- 1 tommy wheel
                           7656 Apr 30 19:33 README.md
                            129 Apr 30 19:33 astyle.sh
-rwxr-xr-x 1 tommy wheel
drwxr-xr-x 19 tommy wheel
                            646 Apr 30 19:33 bench
drwxr-xr-x 4 tommy wheel
                            136 Apr 30 19:33 cmake
drwxr-xr-x 14 tommy wheel
                            476 Apr 30 19:33 example
drwxr-xr-x 3 tommy
                    wheel
                            102 Apr 30 19:33 include
drwxr-xr-x 18 tommy
                    wheel
                            612 Apr 30 19:33 tests
```

CMake is used in spdlog! A commonly used build preparing tool.

### **Build and install spdlog**

Try it out in cmake gui

	🛕 CMake 3.7.1 -	/private/tmp/spdlog-build		
Vhere is the source code:	/private/tmp/spdlog			Browse Source
Vhere to build the binaries:	/private/tmp/spdlog-build			Browse Build
Search:		🗸 Grouped 🔽 Advanc	ed 🕂 Add Entr	y Remove Entry
Name			Value	
Ungrouped Entries				
CMAKE				
CMAKE_AR			/Applicat	tions/Xcode.app/
CMAKE_BUILD_TYPE				
CMAKE_CONFIGURA	TION_TYPES		Debug;R	elease;MinSizeRel
CMAKE_CXX_FLAGS				
CMAKE_CXX_FLAGS	_DEBUG		-g	
CMAKE_CXX_FLAGS Press Configure to	_DEBUG o update and display new values i	n red, then press Generate to gen	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat	_DEBUG o update and display new values i	n red, then press Generate to gen It Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i re Open Project Curren	n red, then press Generate to gen It Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen It Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat	_DEBUG o update and display new values i re Open Project Curren	n red, then press Generate to gen It Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i re Open Project Curren	n red, then press Generate to gen It Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat	_DEBUG o update and display new values i re Open Project Curren	n red, then press Generate to gen It Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i re Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.
CMAKE_CXX_FLAGS Press Configure to Configure Generat Configuring done	_DEBUG o update and display new values i e Open Project Curren	n red, then press Generate to gen t Generator: Xcode	-g erate selected	build files.

```
In [14]:
```

```
%%bash
cd /tmp/spdlog-build
ls -l
```

total 24								
-rw-rr	1	tommy	wheel	16159	Apr	30	19 <b>:</b> 34	CMakeCache.txt
drwxr-xr-x	11	tommy	wheel	374	Apr	30	19 <b>:</b> 34	CMakeFiles
drwxr-xr-x	15	tommy	wheel	510	Apr	30	19 <b>:</b> 34	CMakeScripts
-rw-rr	1	tommy	wheel	2776	Apr	30	19 <b>:</b> 33	DartConfiguration.tc
1								
drwxr-xr-x	3	tommy	wheel	102	Apr	30	19 <b>:</b> 33	Testing
-rw-rr	1	tommy	wheel	2544	Apr	30	19 <b>:</b> 34	cmake_install.cmake
drwxr-xr-x	5	tommy	wheel	170	Apr	30	19 <b>:</b> 34	generated
drwxr-xr-x	3	tommy	wheel	102	Apr	30	19 <b>:</b> 34	spdlog.xcodeproj
drwxr-xr-x	4	tommv	wheel	136	Apr	30	19:33	tests

```
In [15]:
```

```
%%bash
cd /tmp/spdlog-build
xcodebuild -configuration Release -target install
```

```
=== BUILD AGGREGATE TARGET ZERO_CHECK OF PROJECT spdlog WITH CONFIGU RATION Release ===
```

Check dependencies

Write auxiliary files write-file /tmp/spdlog-build/spdlog.build/Release/ZERO\_CHECK.build/S cript-E5A697B66AFA4CA9A63FC2F7.sh chmod 0755 /tmp/spdlog-build/spdlog.build/Release/ZERO\_CHECK.build/S cript-E5A697B66AFA4CA9A63FC2F7.sh

PhaseScriptExecution CMake\ Rules /tmp/spdlog-build/spdlog.build/Rel ease/ZERO\_CHECK.build/Script-E5A697B66AFA4CA9A63FC2F7.sh cd /tmp/spdlog /bin/sh -c /tmp/spdlog-build/spdlog.build/Release/ZERO\_CHECK.bui ld/Script-E5A697B66AFA4CA9A63FC2F7.sh echo ""

make -f /tmp/spdlog-build/CMakeScripts/ReRunCMake.make

#### Notice the installed .cmake files and spdlog.pc

- .cmake Facilitates to use this package from other CMake projects
- spdlog.pc Facilitates to use this package from other projects using the pkg-config utility

### Introduce spdlog in TickCounter project

Use a new branch and name it "logger".

In [16]:

%%bash
cd /tmp/TickProject
git checkout -b logging
# git branch logging
# git checkout logging
Switched to a new branch 'logging'

#### Add some code using spdlog and fix build system

In [17]:

%%bash . quickci Log4 Log5 Log6 git log [logging 38deed5] Fixed Log4 2 files changed, 18 insertions(+), 11 deletions(-) [logging 263f816] Fixed Log5 2 files changed, 11 insertions(+), 5 deletions(-) [logging ea49c43] Fixed Log6 2 files changed, 8 insertions(+), 12 deletions(-) commit ea49c4358bc28ce84d05a08b41eecb1d6109f74d Author: Tommy Carlsson <topcatse@yahoo.se> Sun Apr 30 19:35:06 2017 +0200 Date: Fixed Log6 commit 263f8162e3e9c1a0b16c5ea5e49a1831285f2294 Author: Tommy Carlsson <topcatse@yahoo.se> Sun Apr 30 19:35:06 2017 +0200 Date: Fixed Log5 commit 38deed56b7f659ebc306c0d877e8570e41fdf198 Author: Tommy Carlsson <topcatse@yahoo.se> Sun Apr 30 19:35:06 2017 +0200 Date: Fixed Log4 commit 6845752eb01a38258ee752c90c6caf18813a33e6 Author: Tommy Carlsson <topcatse@yahoo.se> Date: Sun Apr 30 19:30:24 2017 +0200 Initial revision commit 19b738a045a3b7ceefe31ffcfe93020437b64a18 Author: Tommy Carlsson <topcatse@yahoo.se> Date: Sun Apr 30 19:28:41 2017 +0200

Initial revision

Check differences in SourceTree. Build and run.

In [18]:

```
%%bash
cd /tmp/TickProject
ninja -C build
ninja: Entering directory `build'
[0/1] Regenerating build files
The Meson build system
Version: 0.39.1
Source dir: /private/tmp/TickProject/src
Build dir: /private/tmp/TickProject/build
Build type: native build
Project name: TickCounterProject
Native cpp compiler: c++ (clang 8.1.0)
Build machine cpu family: x86 64
Build machine cpu: x86 64
Found pkg-config: /usr/local/bin/pkg-config (0.29.2)
Native dependency spdlog found: YES 1.0.0
Dependency threads found: YES
Build targets in project: 1
[1/2] Compiling cpp object 'tickcounter@exe/main.cpp.o'
[2/2] Linking target tickcounter
clang: warning: argument unused during compilation: '-pthread' [-Wun
used-command-line-argument]
```

### Merge logging branch into master

There are different conceps involved in a merge or could be.

- Fast forward
- Three-way merge
- Rebase
- Conflicts



#### **Fast forward**



Three-way merge





Rebase (change the history)









### Thou shalt not rebase commits that you have pushed to a remote repository.

### Merge conflict

Let's create a conflict on the last line in main.cpp

Also perform development on a another branch. Let's call it features.

```
In [19]:
```

%%bash cd /tmp/TickProject/src git checkout master git branch features

Switched to branch 'master'

In [20]:

```
%%bash
cd /tmp/TickProject/src
echo This row at the end will prevent TickProject from building >> main.cpp
git diff
git commit -am "This will be a conflict"
diff --git a/src/main.cpp b/src/main.cpp
index ldfelbc..a0c98f7 100644
--- a/src/main.cpp
+++ b/src/main.cpp
@@ -40,3 +40,4 @@ int main(int argc, char* argv[])
return 0;
}
+This row at the end will prevent TickProject from building
[master 12b6581] This will be a conflict
1 file changed, 1 insertion(+)
```

Add some features on the features branch

In [21]:

```
%%bash
cd /tmp/TickProject/src
git checkout features
sed -i 's/Hello World/Hello Makers/g' main.cpp
git diff
git commit -am "Changed greeting"
diff --git a/src/main.cpp b/src/main.cpp
index 1dfe1bc..360c171 100644
--- a/src/main.cpp
+++ b/src/main.cpp
00 -24,7 +24,7 00 private:
 int main(int argc, char* argv[])
 {
  cout << "Hello World!" << endl;</pre>
+ cout << "Hello Makers!" << endl;</pre>
   TickReceiver receiver;
[features b5ea177] Changed greeting
 1 file changed, 1 insertion(+), 1 deletion(-)
Switched to branch 'features'
```

#### In [22]:

```
%%bash
cd /tmp/TickProject/src
echo // This row at the end will compile >> main.cpp
git diff
git commit -am "Added a row at the end of main.cpp"
diff --git a/src/main.cpp b/src/main.cpp
index 360c171..cf104bc 100644
--- a/src/main.cpp
+++ b/src/main.cpp
@@ -40,3 +40,4 @@ int main(int argc, char* argv[])
return 0;
}
+// This row at the end will compile
[features fbc1457] Added a row at the end of main.cpp
1 file changed, 1 insertion(+)
```

#### Merge features onto master --> Conflict

Check always out the branch which shall be modified!

```
In [26]:
```

%%bash
cd /tmp/TickProject/src
git checkout master
git merge features

Auto-merging src/main.cpp CONFLICT (content): Merge conflict in src/main.cpp Automatic merge failed; fix conflicts and then commit the result.

Already on 'master'

#### **Resolve conlict in SourceTree**



**Commit merge** 

```
In [27]:
```

%%bash
cd /tmp/TickProject/src
git status

On branch master
All conflicts fixed but you are still merging.
 (use "git commit" to conclude merge)

Changes to be committed:

modified: main.cpp

In [28]:

%%bash
cd /tmp/TickProject/src
git commit -am "Features added"

[master 1fe644d] Features added

#### Rebase

#### This rebase scenario will target these goals:

- The logging branch (spdlog) is merged into the master branch
- An appropriate commit message (i.e, not 'Fixed Log6')
- Only one commit from logging (squash several commits to one)
- In [29]:

```
%%bash
cd /tmp/TickProject
git checkout logging
```

Switched to branch 'logging'

#### I usually create a tmp branch at this point (If something goes wrong, just remove after)

In [30]:

```
%%bash
cd /tmp/TickProject
git branch logging-tmp
```

#### Use git rebase interactively to squash the logging branch

Graph	Description					
•	pmaster Features add					
•	<b>b</b> features	s Added a row at the end of main.cpp				
	Changed greeting					
•	This will be a conflict					
ļ	<b>1</b> logging	logging-tmp	Fixed Log6			
	Fixed Log5					
	Fixed Log4					
	Added gitignore					
	Fixed Log4 Added gitignore					

Use git rebase interactively Let's perform this in SourceTree via a terminal.

Like so: git rebase -i 06558f1d77a09bb41a97bf3eda20e1af3f551a39

```
ick 8565276 Fixed Log4
pick a39993e Fixed Log5
pick 43b0bd0 Fixed Log6
# Rebase f727868..43b0bd0 onto f727868 (3 commands)
#
# Commands:
 p, pick = use commit
#
# r, reword = use commit, but edit the commit message
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
# f, fixup = like "squash", but discard this commit's log message
# x, exec = run command (the rest of the line) using shell
# d, drop = remove commit
# These lines can be re-ordered; they are executed from top to bottom.
# If you remove a line here THAT COMMIT WILL BE LOST.
#
# However, if you remove everything, the rebase will be aborted.
#
 Note that empty commits are commented out
#
```

After squash

Q	logging Introduced spdlog						
•	<b>1</b> master Features add						
│ │ │ ∳	features Added a row at the end of main.cpp						
	Changed greeting						
	This will be a conflict						
•	<pre> logging-tmp Fixed Log6 </pre>						
	Fixed Log5						
	Fixed Log4						
	Added gitignore						
	Add						
	Initial revision						

#### In [31]:

%%bash
cd /tmp/TickProject
git checkout logging
git rebase master

First, rewinding head to replay your work on top of it... Applying: Introduced spdlog Using index info to reconstruct a base tree... M src/main.cpp Falling back to patching base and 3-way merge... Auto-merging src/main.cpp CONFLICT (content): Merge conflict in src/main.cpp Patch failed at 0001 Introduced spdlog The copy of the patch that failed is found in: .git/rebase-apply/pat ch

When you have resolved this problem, run "git rebase --continue". If you prefer to skip this patch, run "git rebase --skip" instead. To check out the original branch and stop rebasing, run "git rebase --abort".

Already on 'logging' .git/rebase-apply/patch:32: trailing whitespace. .git/rebase-apply/patch:34: trailing whitespace. .git/rebase-apply/patch:53: trailing whitespace. .git/rebase-apply/patch:65: trailing whitespace. spdlog\_inc = include\_directories(spdlog\_dep.get\_pkgconfig\_variable(' includedir')) warning: 4 lines add whitespace errors.

error: Failed to merge in the changes.

#### After rebase of logging



#### Now master can be updated with a fast forward merge of logging

In [32]:

#### Remove logging-tmp

In [33]:

%%bash cd /tmp/TickProject git branch -d logging-tmp

```
error: The branch 'logging-tmp' is not fully merged.
If you are sure you want to delete it, run 'git branch -D logging-tm
p'.
```

In [34]:

```
%%bash
cd /tmp/TickProject
git branch -D logging-tmp
```

Deleted branch logging-tmp (was ea49c43).

#### Reflog to your help if something has gone really wrong

In [35]:

```
%%bash
cd /tmp/TickProject
git reflog
```

```
lec464f HEAD@{0}: merge logging: Fast-forward
1fe644d HEAD@{1}: checkout: moving from logging to master
lec464f HEAD@{2}: rebase finished: returning to refs/heads/logging
lec464f HEAD@{3}: commit: Added logging
1fe644d HEAD@{4}: rebase: checkout master
a605e52 HEAD@{5}: checkout: moving from logging to logging
a605e52 HEAD@{6}: rebase -i (finish): returning to refs/heads/loggin
g
a605e52 HEAD@{7}: rebase -i (fixup): Introduced spdlog
aelfeae HEAD@{8}: rebase -i (fixup): # This is a combination of 2 co
mmits.
4c4c9bf HEAD@{9}: rebase -i (reword): Introduced spdlog
38deed5 HEAD@{10}: cherry-pick: fast-forward
6845752 HEAD@{11}: rebase -i (start): checkout 6845752eb01a38258ee75
2c90c6caf18813a33e6
ea49c43 HEAD@{12}: checkout: moving from master to logging
1fe644d HEAD@{13}: commit (merge): Features added
7596e60 HEAD@{14}: checkout: moving from master to master
7596e60 HEAD@{15}: commit: Added gitignore
12b6581 HEAD@{16}: reset: moving to 12b6581c08cf794e6152e5b11428fdb6
44515822
69c6fcb HEAD@{17}: commit: Merge branch 'features'
fc0826a HEAD@{18}: commit (merge): Added gitignore
12b6581 HEAD@{19}: checkout: moving from features to master
fbc1457 HEAD@{20}: commit: Added a row at the end of main.cpp
b5ea177 HEAD@{21}: commit: Changed greeting
6845752 HEAD@{22}: checkout: moving from master to features
12b6581 HEAD@{23}: commit: This will be a conflict
6845752 HEAD@{24}: checkout: moving from logging to master
ea49c43 HEAD@{25}: commit: Fixed Log6
263f816 HEAD@{26}: commit: Fixed Log5
38deed5 HEAD@{27}: commit: Fixed Log4
6845752 HEAD@{28}: checkout: moving from master to logging
6845752 HEAD@{29}: commit: Initial revision
19b738a HEAD@{30}: commit (initial): Initial revision
```

### Some more Git concepts

- Detached HEAD HEAD pointes to a commit (not a good state to be in)
- Hunk stage and/or discard changes of a file
- Amend fix last commit code and/or message
- Stash put changes in the stash
- Blame check who did what in a file
- Cherry pick get files from a commit

#### !. quickedit v0.2

Staged files	•••
Verify src/main.cpp +++ Hunk 1: Lines 15-21 Stage hunk Discard h	unk
16 16	
17 17 ++m_counter;	
<pre>18 - spdlog::get("console")-&gt;info("{}", m_counter);</pre>	
18 + cout << m_counter;	
19 19 }	
20 20	
21 21 private	
Unstaged files	
Hunk 2 : Lines 24-42 Stage hunk Discard h	unk
src/main.cpp ••• 24.24	
src/meson.build ••• 25 25 int main(int argc, char* argv[])	
<pre>27 - auto console = spdlog::stdout_logger_st("console");</pre>	
28 -	
<pre>29 - console-&gt;warn("Hello World!");</pre>	
30 -	
31 - TickReceiver receiver;	
27 + cout << "Hello World!" << endl:	

### GitHub

- 1. Create a repository on your GitHub account @ https://github.com (https://github.com)
- 2. Bind your local account to the GitHub upstream repository
- 3. Clone upstream repository to a second repository
- 4. Add some code on a new branch travis and push upstream
- 5. On secondary repository, fetch from upstream and merge or do a pull
- 6. Create a pull request

#### 2. Bind

#### In [40]:

%%bash cd /tmp/TickProject git checkout master git remote add origin https://github.com/topcatse/TickProject.git git push --set-upstream origin master M src/main.cpp

M src/meson.build Branch master set up to track remote branch master from origin. Already on 'master' Everything up-to-date

#### 3. Clone TickProject from GitHub to a second repository

In [41]:

```
%%bash
cd /tmp
git clone https://github.com/topcatse/TickProject.git TickProject2
cd TickProject2
tree -aCI .git
```

```
    → .gitignore
    → src
    → KeyTicker.h
    → TickerInterface.h
    → main.cpp
    → meson.build
```

1 directory, 5 files

```
Cloning into 'TickProject2'...
```

In [42]:

%%bash cd /tmp ls -l

total 5976								
drwxr-xr-x	4	tommy	wheel	136	Apr	30	19 <b>:</b> 34	3rd
drwxr-xr-x	6	tommy	wheel	204	Мау	1	22 <b>:</b> 56	TickProject
drwxr-xr-x	5	tommy	wheel	170	Мау	1	23:01	TickProject
2								
drwx	3	tommy	wheel	102	Apr	24	20:58	<pre>com.apple.l</pre>
aunchd.7DNtr	1Fy	yaN						
drwx	3	_mbsetupuser	wheel	102	Apr	24	20:58	<pre>com.apple.l</pre>
aunchd.NxV31	8W4	45z						
drwx	3	_mbsetupuser	wheel	102	Apr	24	20:58	<pre>com.apple.l</pre>
aunchd.YA0N0	Q61	VN1						
drwx	3	tommy	wheel	102	Apr	24	20:58	<pre>com.apple.l</pre>
aunchd.nrF2E	Bt7E	BQG						
drwx	5	tommy	wheel	170	Apr	24	21:04	ctr-2714706
87								
drwx	5	tommy	wheel	170	Apr	24	21:04	ctr-8533333
16								
drwxr-xr-x	15	tommy	wheel	510	Apr	30	19 <b>:</b> 33	spdlog
drwxr-xr-x@	12	tommy	wheel	408	Apr	30	19 <b>:</b> 34	<pre>spdlog-buil</pre>
d								
-rw-rr	1	root	wheel	2019478	Apr	25	16 <b>:</b> 50	wifi-04-25-
201716:50:	03.	.log						
-rw-rr	1	root	wheel	2048000	Apr	29	15 <b>:</b> 49	wifi-04-29-
201715:49:	54.	.log						
-rw-rr	1	root	wheel	2048000	Apr	29	18 <b>:</b> 56	wifi-04-29-
201718:56:	04.	.log						

#### 4. Add some code locally in a new branch 'travis' and push upstream

```
In [43]:
```

```
%%bash
cd /tmp/TickProject
git checkout -b travis
git archive --format=tar --remote=~/src/maker-presentation v0.2 .travis.yml | tar
git add .travis.yml
git commit -am "Added travis-ci build"
git push -u origin travis
```

```
M src/main.cpp
M src/meson.build
[travis ebf44f0] Added travis-ci build
3 files changed, 57 insertions(+), 22 deletions(-)
create mode 100644 .travis.yml
Branch travis set up to track remote branch travis from origin.
Switched to a new branch 'travis'
To https://github.com/topcatse/TickProject.git
(https://github.com/topcatse/TickProject.git)
* [new branch] travis -> travis
```

#### 5. Fetch from upstream to TickProject2

In [44]:

```
%%bash
cd /tmp/TickProject2
git fetch origin
git branch -a
```

```
* master
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/travis
```

```
From https://github.com/topcatse/TickProject
(https://github.com/topcatse/TickProject)
 * [new branch] travis -> origin/travis
```

In [45]:

```
%%bash
cd /tmp/TickProject2
# git checkout -b travis --track origin/travis
git checkout travis
```

Branch travis set up to track remote branch travis from origin.

Switched to a new branch 'travis'

6. At GitHub create a pull request to merge travis into master. Acknowledge. Update TickCounter2!

```
In [46]:
```

```
%%bash
cd /tmp/TickProject2
git fetch origin
git branch -a
```

master

```
* travis
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/travis
```

From https://github.com/topcatse/TickProject
(https://github.com/topcatse/TickProject)
 lec464f..47ca2d3 master -> origin/master

In [47]:

%%bash cd /tmp/TickProject2 git checkout master git log master..origin/master Your branch is behind 'origin/master' by 2 commits, and can be fastforwarded. (use "git pull" to update your local branch) commit 47ca2d3d7edd2db19fe0e0e4f279e211c17ceb9c Merge: lec464f ebf44f0 Author: Tommy Carlsson <topcatse@yahoo.se> Mon May 1 23:03:06 2017 +0200 Date: Merge pull request #1 from topcatse/travis Added travis-ci build commit ebf44f0da3b41cb5c7458af2f7530ff4f93e7b5a Author: Tommy Carlsson <topcatse@yahoo.se> Mon May 1 23:01:47 2017 +0200 Date: Added travis-ci build

Switched to branch 'master'

```
In [48]:
```

```
%%bash
cd /tmp/TickProject2
git merge origin/master
git status
Updating lec464f..47ca2d3
Fast-forward
.travis.yml
               src/main.cpp
src/meson.build
                6 +-----
3 files changed, 57 insertions(+), 22 deletions(-)
create mode 100644 .travis.yml
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean
```

### **Continuous integration**

- Jenkins
- TeamCity
- GitLab
- Bamboo
- Travis-Cl