Getting Started with

Source Control

and

Git

Chris Hunt

@thisisthechris

http://thisisthechris.co.uk/

Nick Charlton

@nickcharlton

http://nickcharlton.net/

Source Control

Keep track of files.

Collaborate on the same project.

Know who to blame.

Git

Distributed.

Modern.

Cool.

FORGET ANYTHING ELSE YOU KNOW.

Concepts

Just a bunch of snapshots & labels.

Commit

Branch

Tag

HEAD

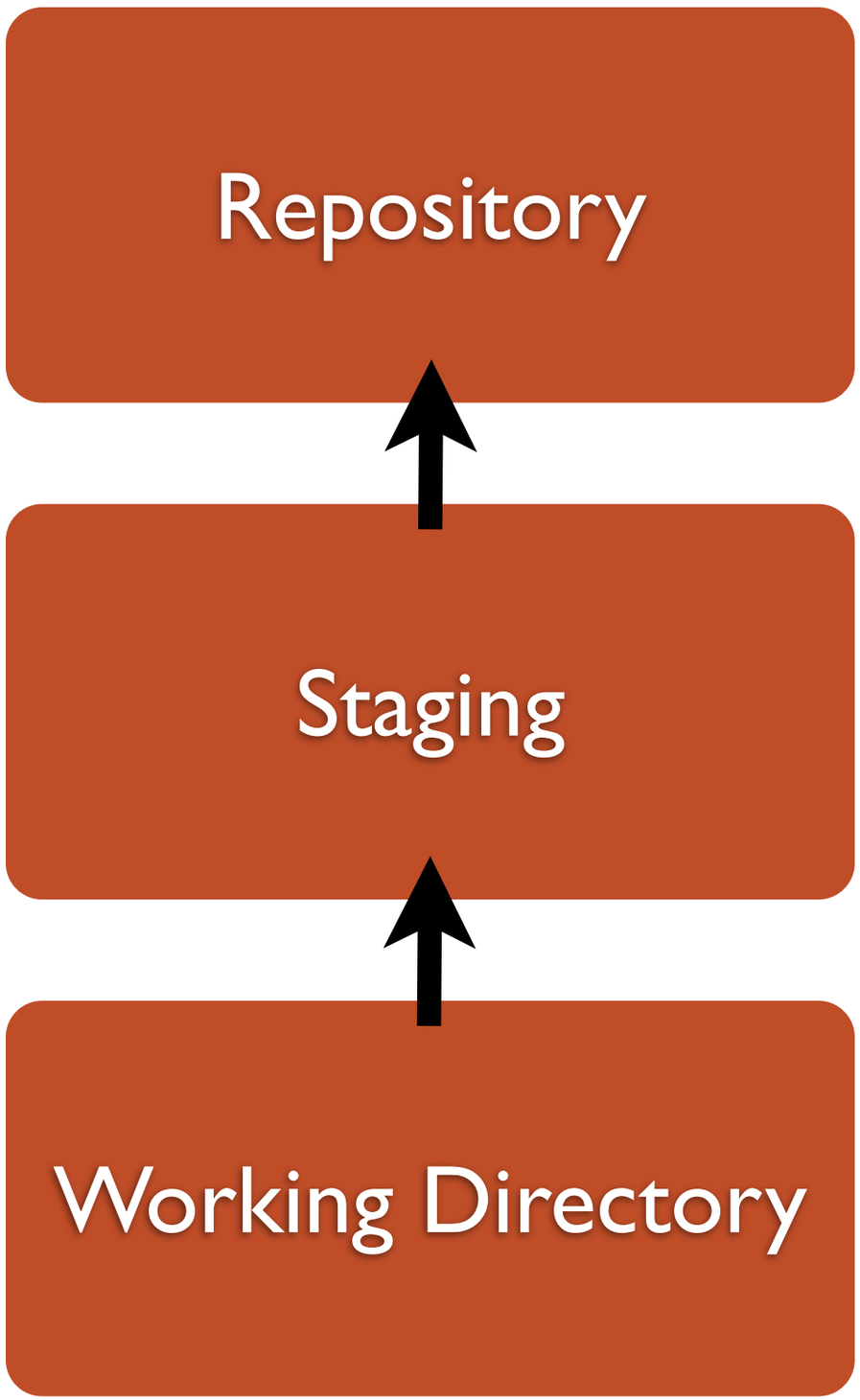
Getting Git

Debian/Ubuntu: `# apt-get install git`

Mac (or Xcode): `$ brew install git`

Windows: `mysysgit`

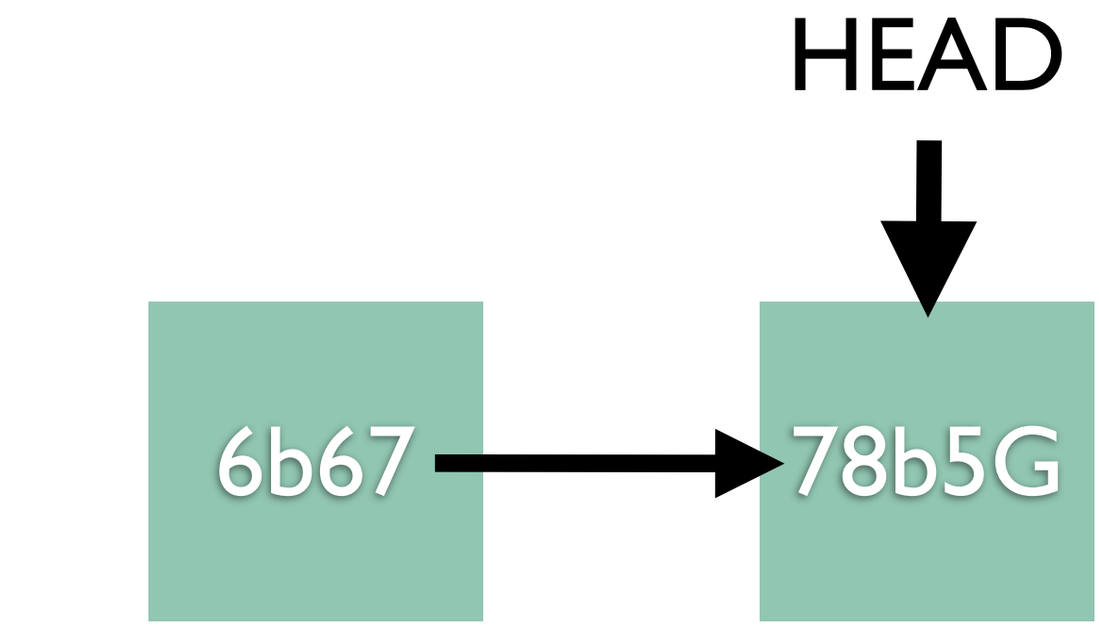
Committing




```
git commit -m ‘Blargh’
```

```
git add file.c
```

```
vim file.c
```



“Initial Commit.” “Blargh”

$ git log

```
commit be0bb1ed7a722cfd569494d1ee2d7f22164a6019
Author: Nick Charlton <hello@nickcharlton.net>
Date:   Sun Feb 26 16:41:43 2012 +0000

    Commit message..

commit dcc5c23c9a3a391461070751bf739b648605eec9
Author: Nick Charlton <hello@nickcharlton.net>
Date:   Sun Feb 26 14:22:31 2012 +0000

    Commit message.

commit 6f9a28631ab8ce9c8aa29039f538c36b8308791e
Author: Nick Charlton <hello@nickcharlton.net>
Date:   Sun Feb 26 14:09:34 2012 +0000

    A very long commit message. Somewhere around 75 chars.

commit 86d9f3c21f1720582f59a8fdd38279968c69a9da
Author: Nick Charlton <hello@nickcharlton.net>
Date:   Sun Feb 26 11:07:20 2012 +0000

    You can also use multiple lines.

    Then you can fit in the why.
```

Messages

~~`Fixed stuff.`~~

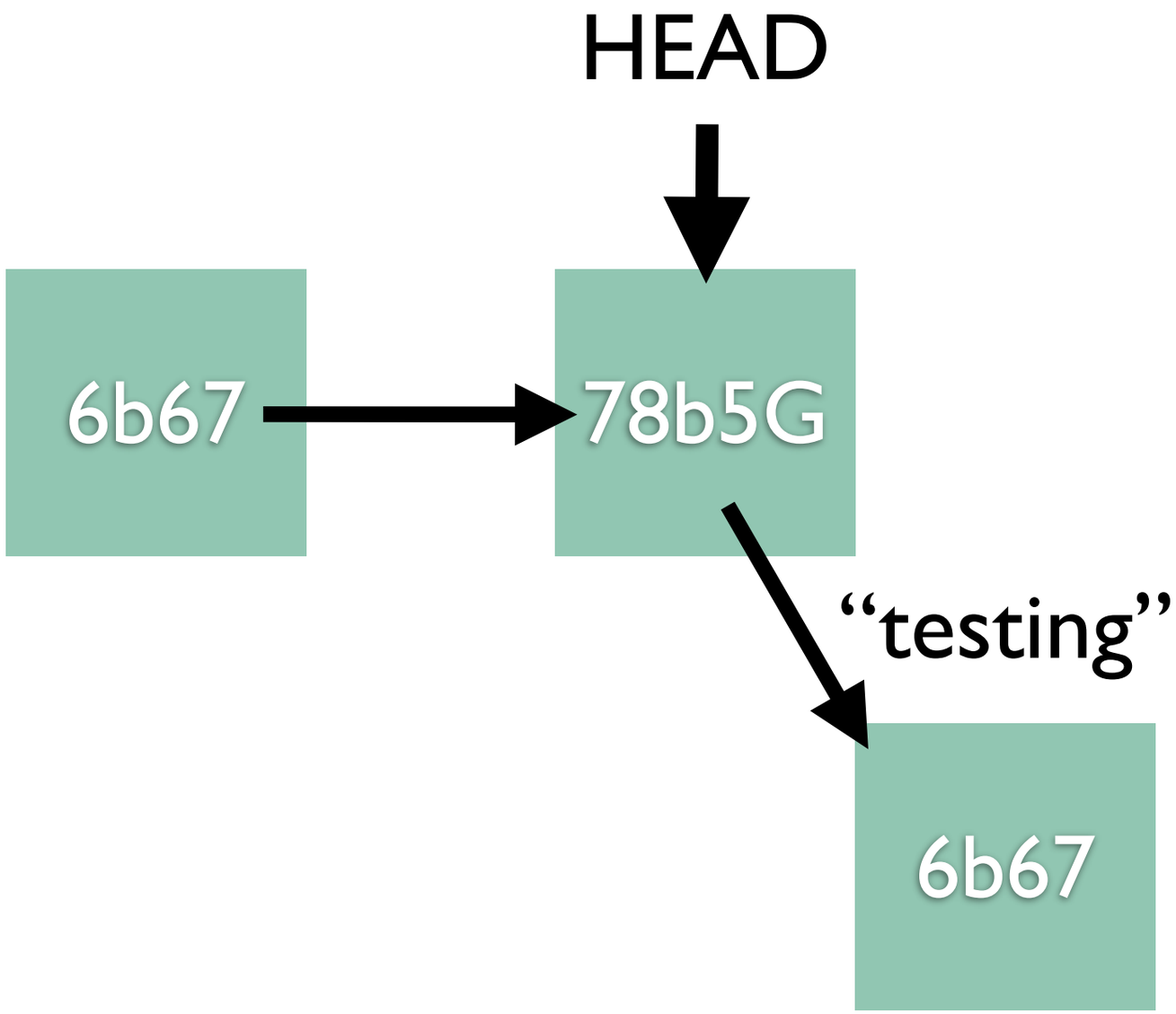
```
Fixes loading bug.

* View wasn’t instantiated correctly.
* Need to alloc before using.
```

Branching

Split off from your main code base.
Implement new features.
Fix bugs.
Try things out.
Practically instantaneous.

```
$ git branch new_branch
$ git checkout new_branch
```

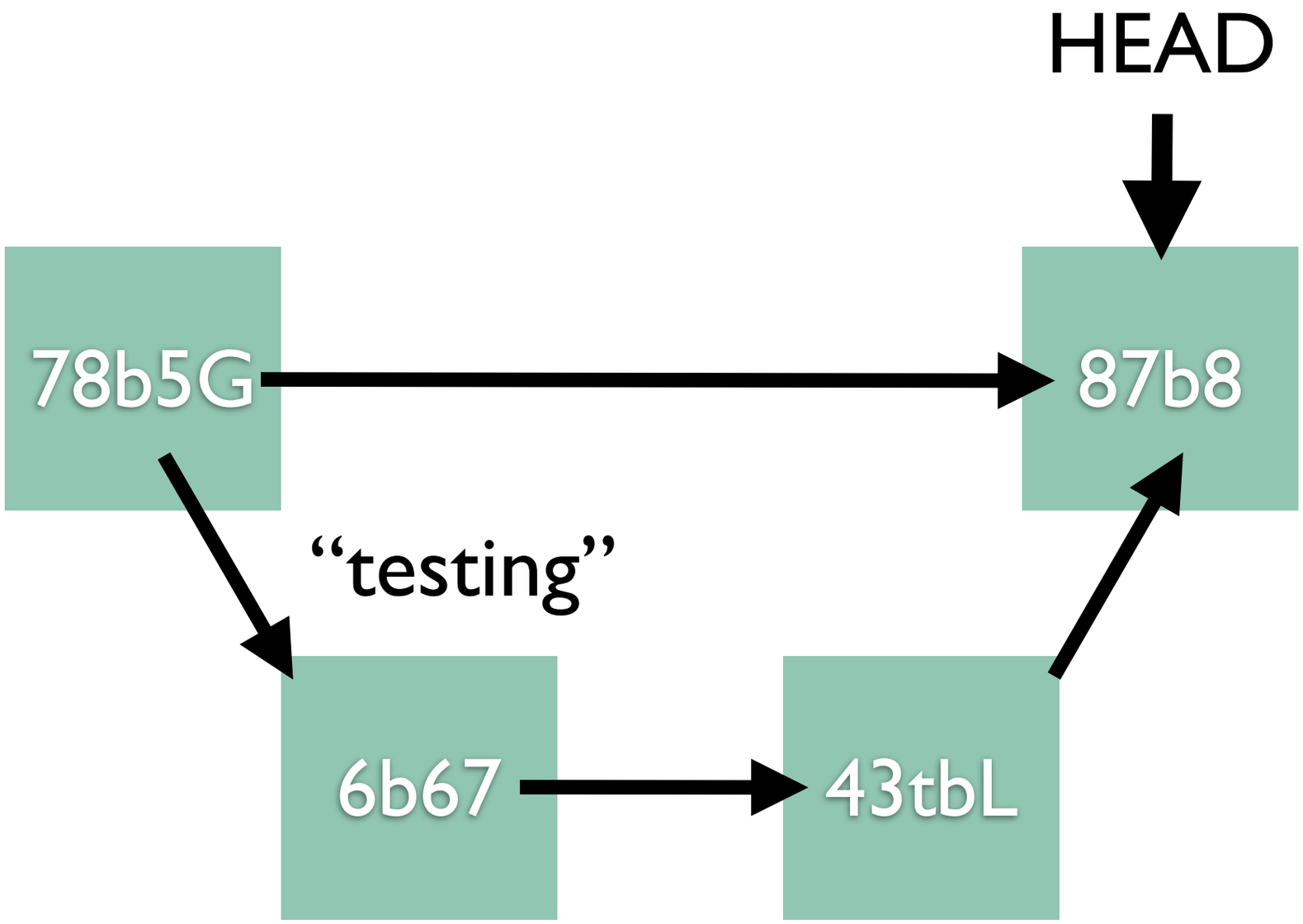
```
$ git checkout -b new_branch
```


HEAD
6b67
78b5G
“testing”
6b67

Merging

Join branches together.
Usually completely automatic.
Pretty damn fast, too.

```
$ git checkout master
$ git merge new_branch
```


HEAD
78b5G
87b8
“testing”
6b67
43tbL

Remotes

Collaborate.
Back up.
Even deploy.

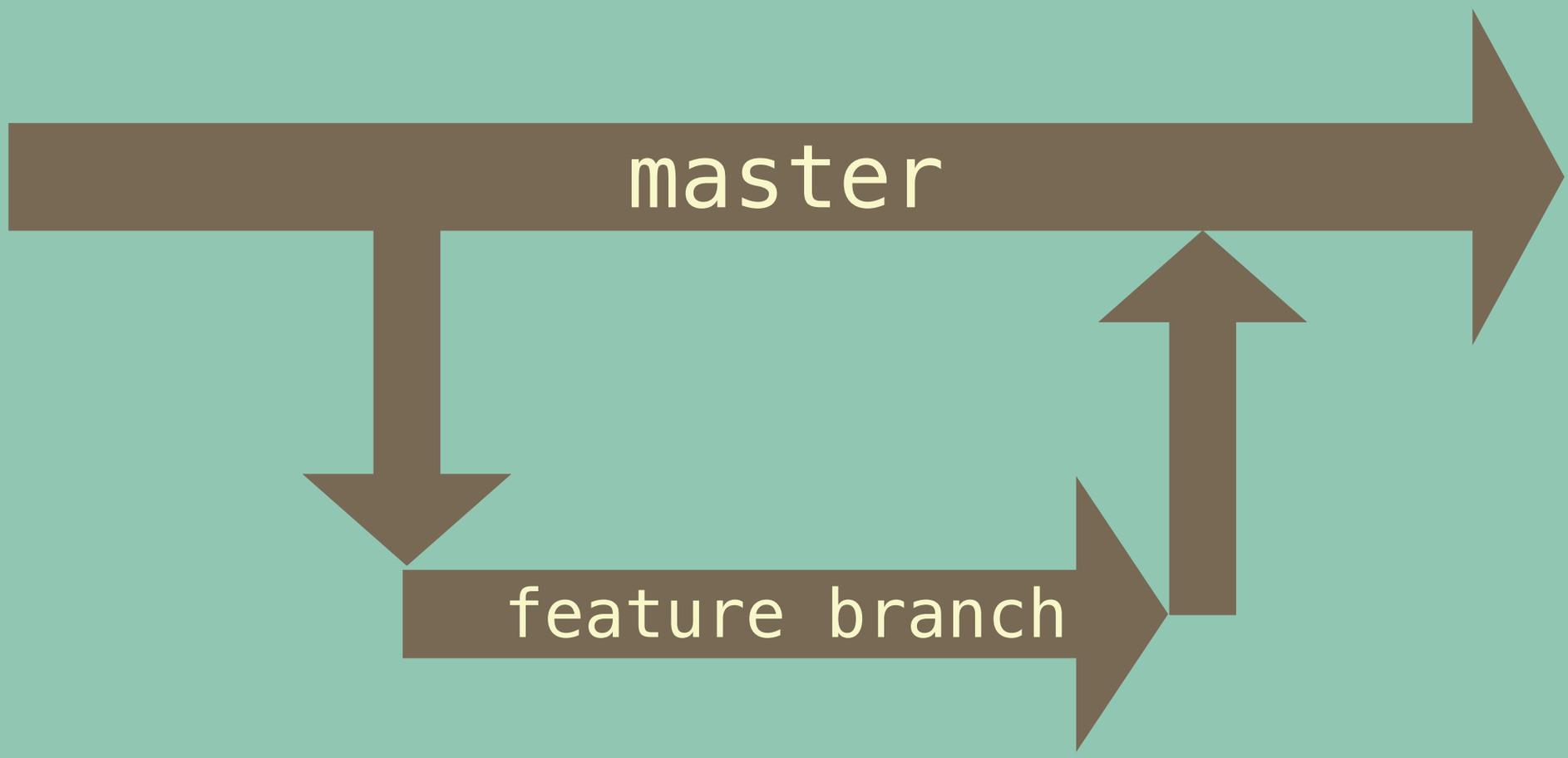
```
$ git remote add origin <url>
$ git push origin master
```

Tagging

Ear mark commits.
Handy for version numbers.
Or other milestones.

```
$ git tag v1.0
```

A Workflow



More

- Rebase
- Submodules
- ???

Resources

- Short Guide: http://url/
- Pro Git Book: http://progit.org/
- Git Reference: http://gitref.org/
- GitHub: https://github.com/

Slides (et. al.) licensed under Creative Commons Attribution-ShareAlike 3.0.

Thanks.