```
$> git init
```

GIT

- What?
- Why?
- How?

GIT - What?

- A software distributed under the terms of the GNU.
- Free
- Distributed version control system

GIT- Why?
What is Versioning Control?

Version control system - to help developers to work together and maintain a complete history of their work

- Allow developers to work simultaneously
- Does not allow to overwriting each other’s changes
- Maintains history of every version
- Two types of VCS system: SVN and GIT

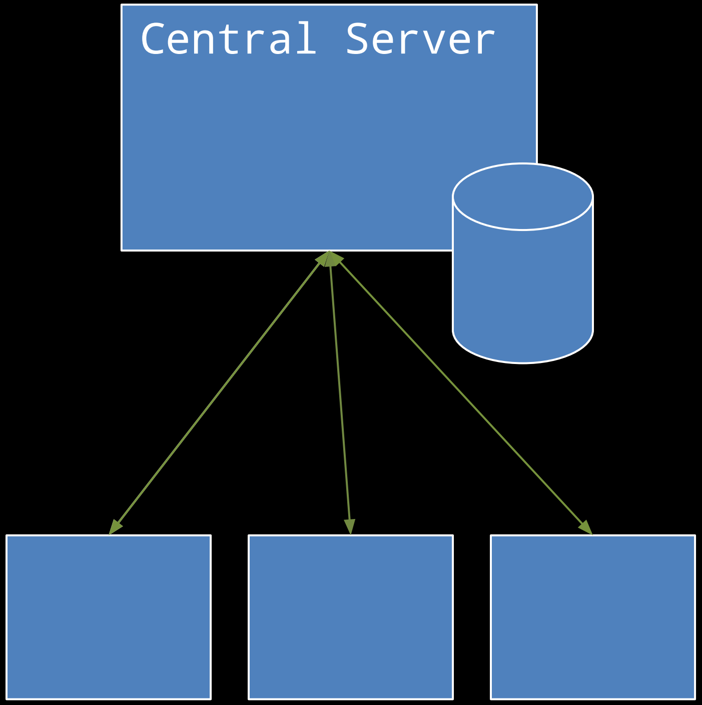
SVN

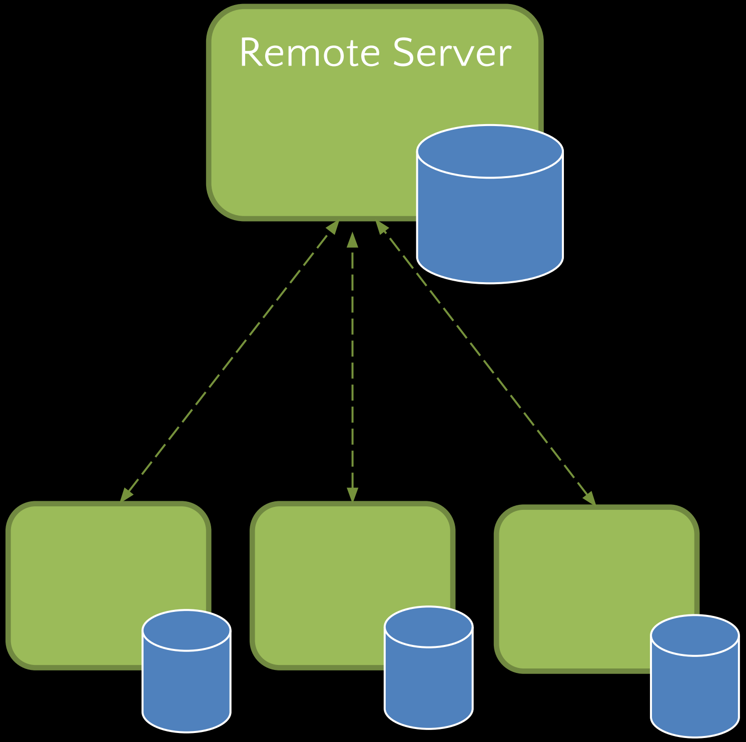
- Centralized
- Checkouts
- Branching
- Conflict Resolution
- No Offline support
- Simple and easy to learn

GIT

- DeCentralized
- Fast, Reliable, Consistent
- Branching / Checkouts
- Conflict Resolutions
- Offline support
- Wide range of deployment hooks

Centralized VC vs. Distributed VC





SVN Commands

co - checkout

list

add

commit

update

Git Commands

init

clone

status

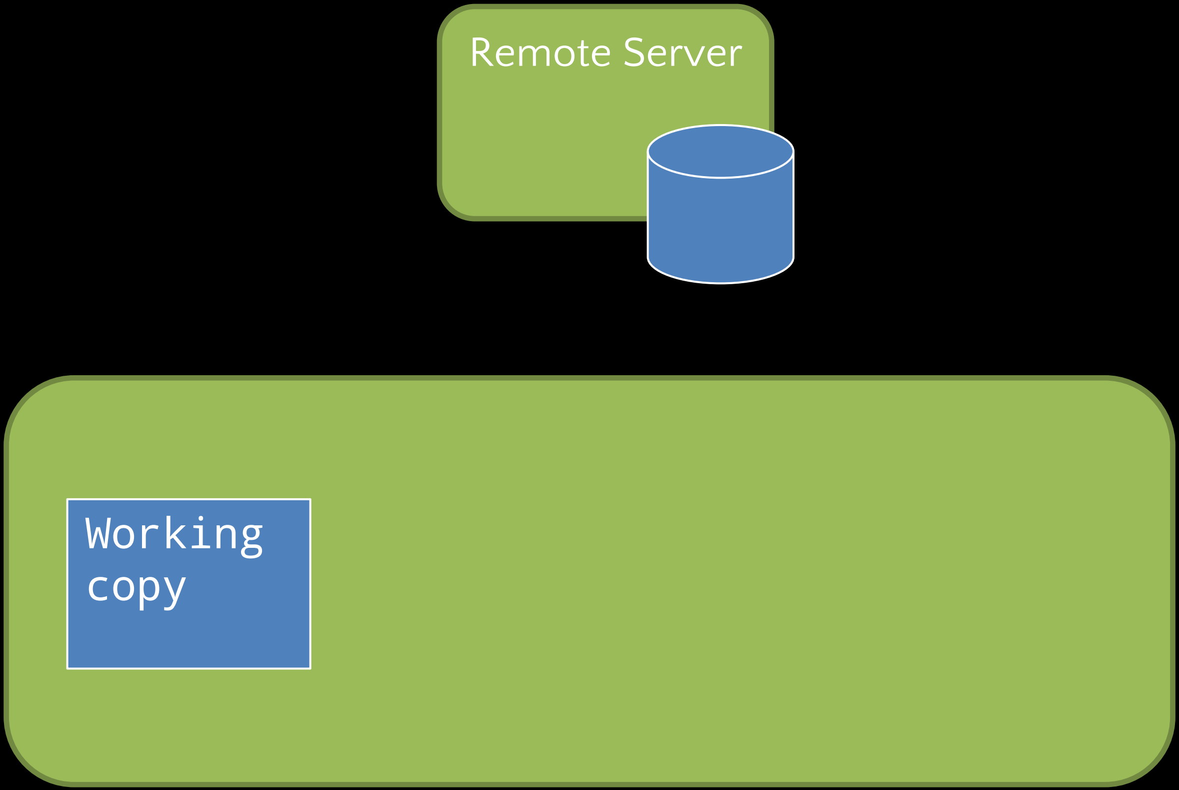
add

commit

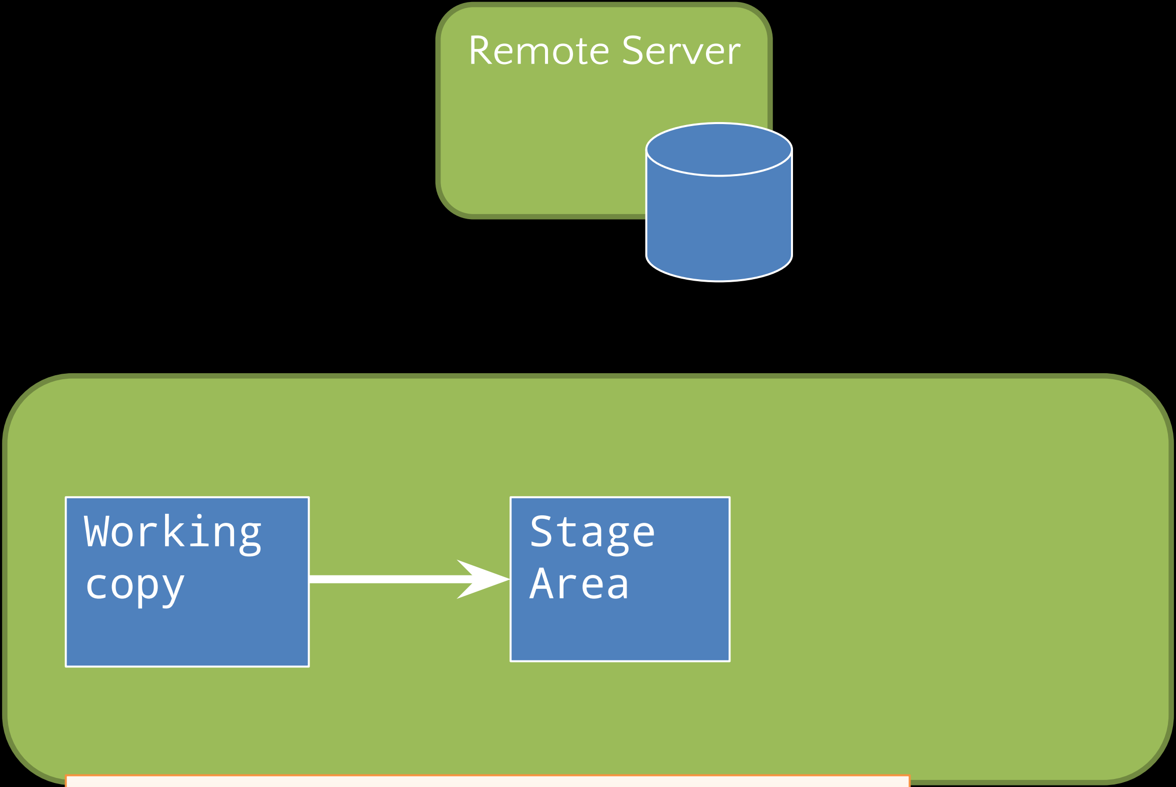
push

pull / fetch

Git Work-flow

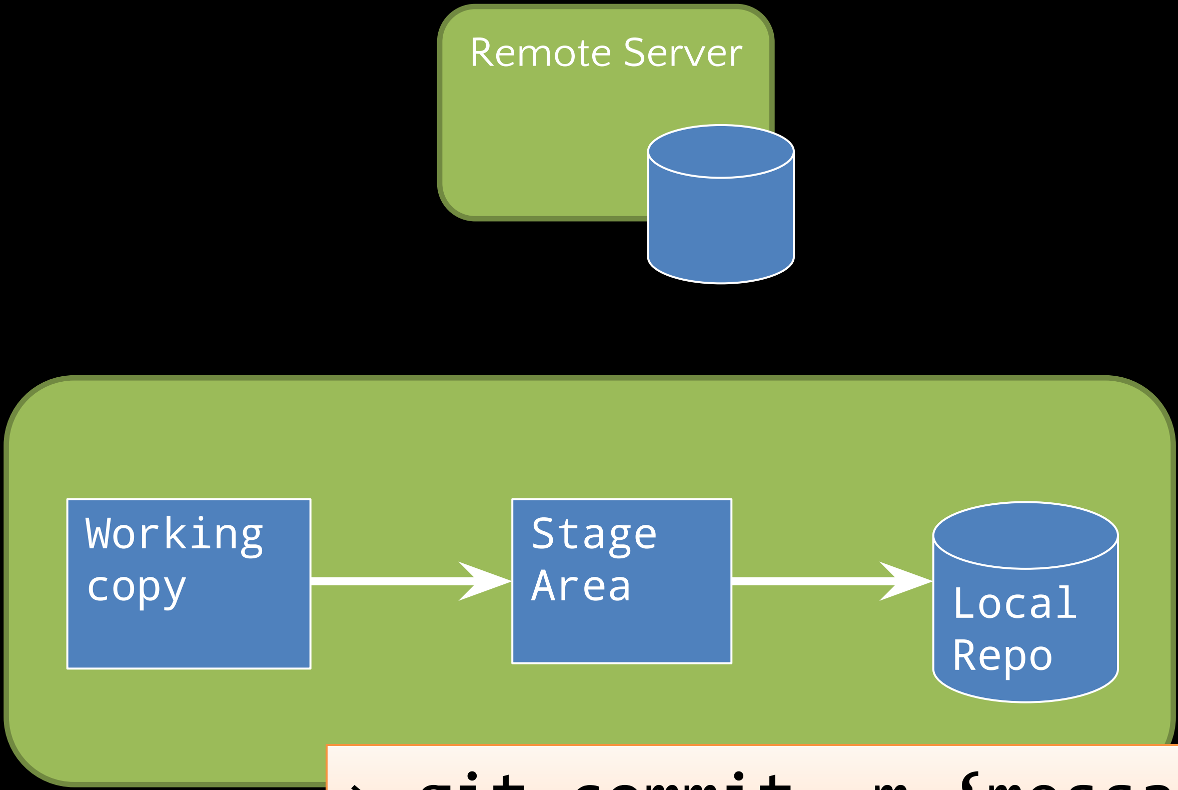


Git Work-flow



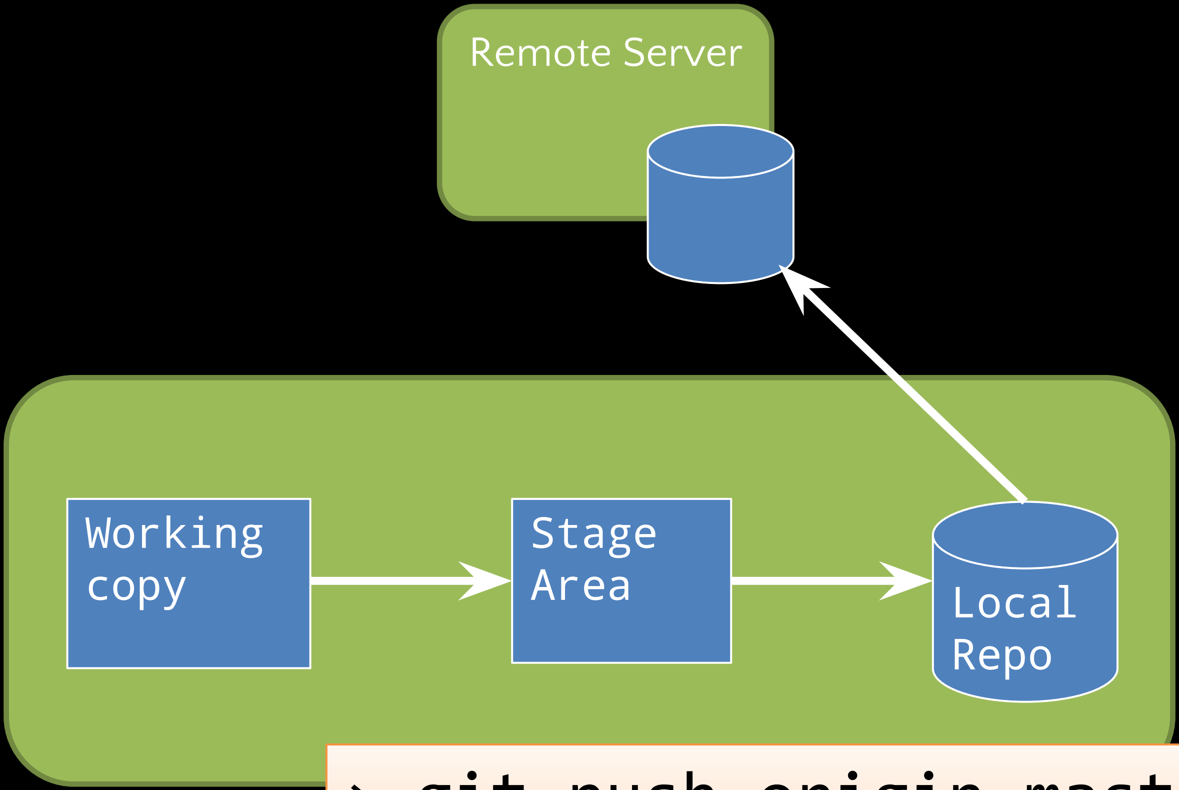
```
> git add file-name
```

Git Work-flow



```
> git commit -m ‘message’
```

Git Work-flow



```
> git push origin master
```

GIT - How?

- Windows
  http://git-scm.com/download/win
- Linux

```
$ sudo apt-get install git
$ yum install git
```

- Mac
  http://git-scm.com/download/mac

Cloning a repository

```
> git clone url

> git clone git@github.com/username/myproject.git

> git clone https://github.com/username/myproject.git

> cd myproject
```

Git config

Global configuration

```
> git config --global user.name your.name
> git config --global user.email your.email
```

Project configuration

```
> cd myproject
> git config user.name your.name
> git config user.email your.email
```

Creating a new Repo

```
> mkdir myproject
> cd myproject
> git init
```

```
Initialized empty Git repository in ~/CoolProject/.git
```

```
> touch README.txt
> git add .
> git commit -m 'my first commit'
```

```
[master (root-commit) 7106a52] my first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.txt
```

Adding a remote server

```
> git remote add origin git@github.com/username/myproject.git
> git remote -v
origin  git@github.com/username/myproject.git (fetch)
origin  git@github.com/username/myproject.git (push)
```

```
> git remote add production git@github.com/live/myproject.git
> git remote -v
origin      git@github.com/username/myproject.git (fetch)
origin      git@github.com/username/myproject.git (push)
production  git@github.com/live/project.git (fetch)
production  git@github.com/live/project.git (push)
```

Adding files

```
> git add [file-name | dot(.)]
```

Will stage your file to be committed
Conventionally `git add .` should be avoided

```
> git add -u
```

Will stage all your deleted files to be deleted from repository

Commit files

```
> git commit
```

Will commit your file to your local repository

```
> git commit -m ‘message’
```

With inline message

```
> git commit -am ‘message’
```

Equivalent of [git add . + git commit -m]

Pull

```
> git pull remote branch
```

Will pull latest commits from the remote repository and try to auto merge them

```
> git pull origin master
```

Push

```
> git push remote branch
```

Will push your commits to the remote repository

```
> git push origin master
```

```
> git push -f origin master
```

Overrides any previous commit/changes

Branching

- Branches in Git basically is a collection of commits
- By default ‘master’ branch is available in all repositories

```
> git branch
```

lists all branches

```
> git branch new-branch
```

creates new branch

```
> git branch -D your-branch
```

deletes a branch

More git commands

```
> git diff [file-name]
```

Shows difference between working copy and last revision

```
> git log
```

Shows a complete history of commits with dates and author-name

More git commands

```
> git blame [file-name]
```

Shows who/when made changes to a file line-by-line

More git commands

```
> git fetch
```

- Remote branch tracking
- Updates your local repository with remote branch

More git commands

```
> git stash
```

Stashing takes the dirty state of your working directory - that is, your modified tracked files and staged changes - and saves it on a stack of unfinished changes that you can reapply at any time.

```
> git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   index.html
#
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be
committed)
#
#       modified:   lib/simplegit.rb
```

```
> git stash
Saved working directory and index state \
  "WIP on master: 049d078 added the index file"
HEAD is now at 049d078 added the index file
(To restore them type "git stash apply")
```

```
> git status
# On branch master
nothing to commit, working directory clean
```

```
> git merge issue53
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit
the result.
```

```
> git merge issue53
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit
the result.
```

Merge Conflict

```
<<<<<<< HEAD:index.html
<div id="footer">contact : email.support@github.
com</div>
=======
<div id="footer">
 please contact us at support@github.com
</div>
>>>>>>> issue53:index.html
```

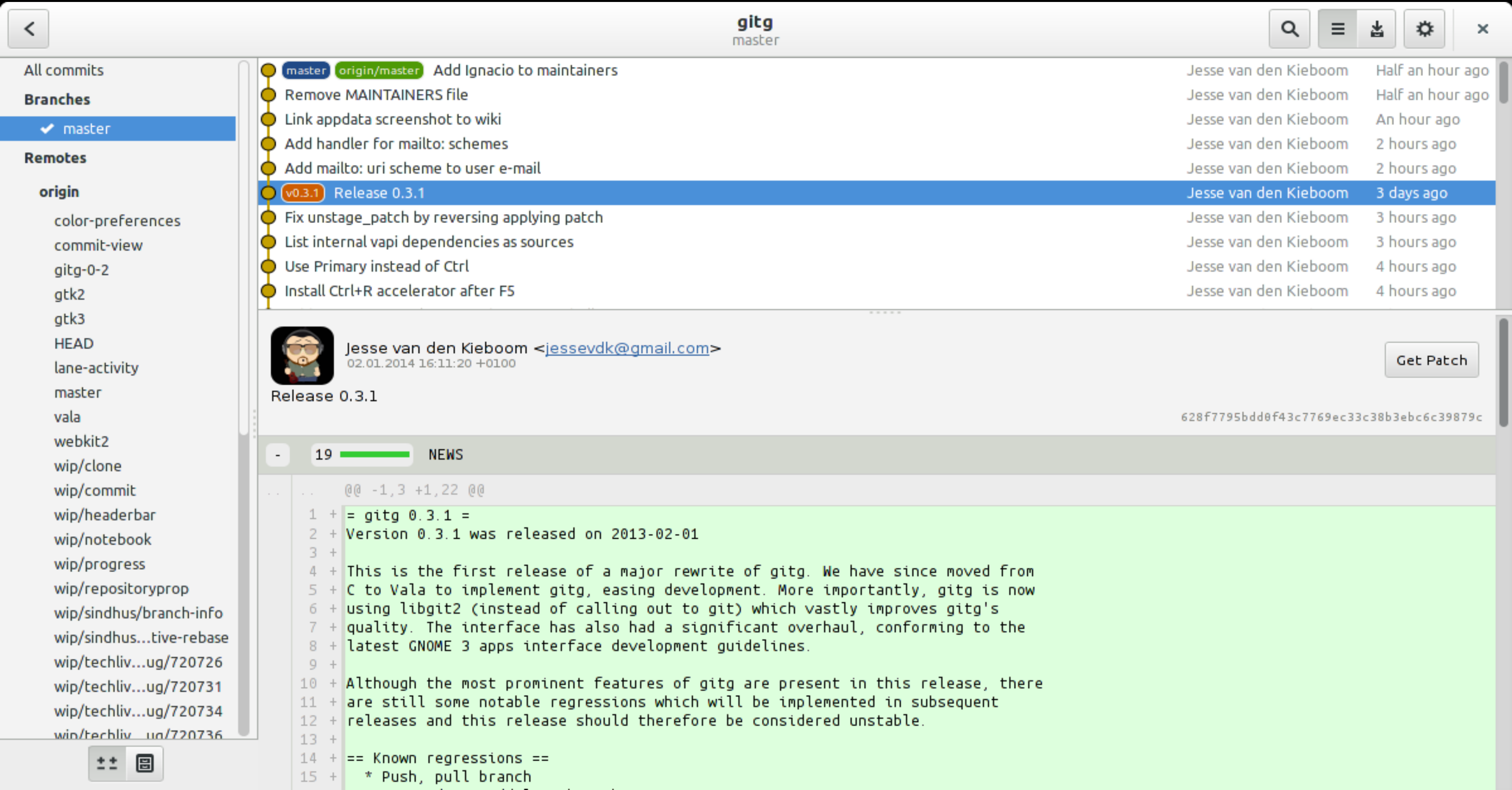
```
<div id="footer">
please contact us at email.support@github.com
</div>
```

Tools

- Following are few very necessary tool for working with git

```
> sudo apt-get install gitg
> sudo apt-get install git-gui
> sudo apt-get install meld
```

GitG (gitg)



master origin/master Add Ignacio to maintainers	Jesse van den Kieboom	Half an hour ago
Remove MAINTAINERS file	Jesse van den Kieboom	Half an hour ago
Link appdata screenshot to wiki	Jesse van den Kieboom	An hour ago
Add handler for mailto: schemes	Jesse van den Kieboom	2 hours ago
Add mailto: uri scheme to user e-mail	Jesse van den Kieboom	2 hours ago
v0.3.1 Release 0.3.1	Jesse van den Kieboom	3 days ago
Fix unstage_patch by reversing applying patch	Jesse van den Kieboom	3 hours ago
List internal vapi dependencies as sources	Jesse van den Kieboom	3 hours ago
Use Primary instead of Ctrl	Jesse van den Kieboom	4 hours ago
Install Ctrl+R accelerator after F5	Jesse van den Kieboom	4 hours ago

Jesse van den Kieboom <jessevdk@gmail.com>
02.01.2014 16:11:20 +0100

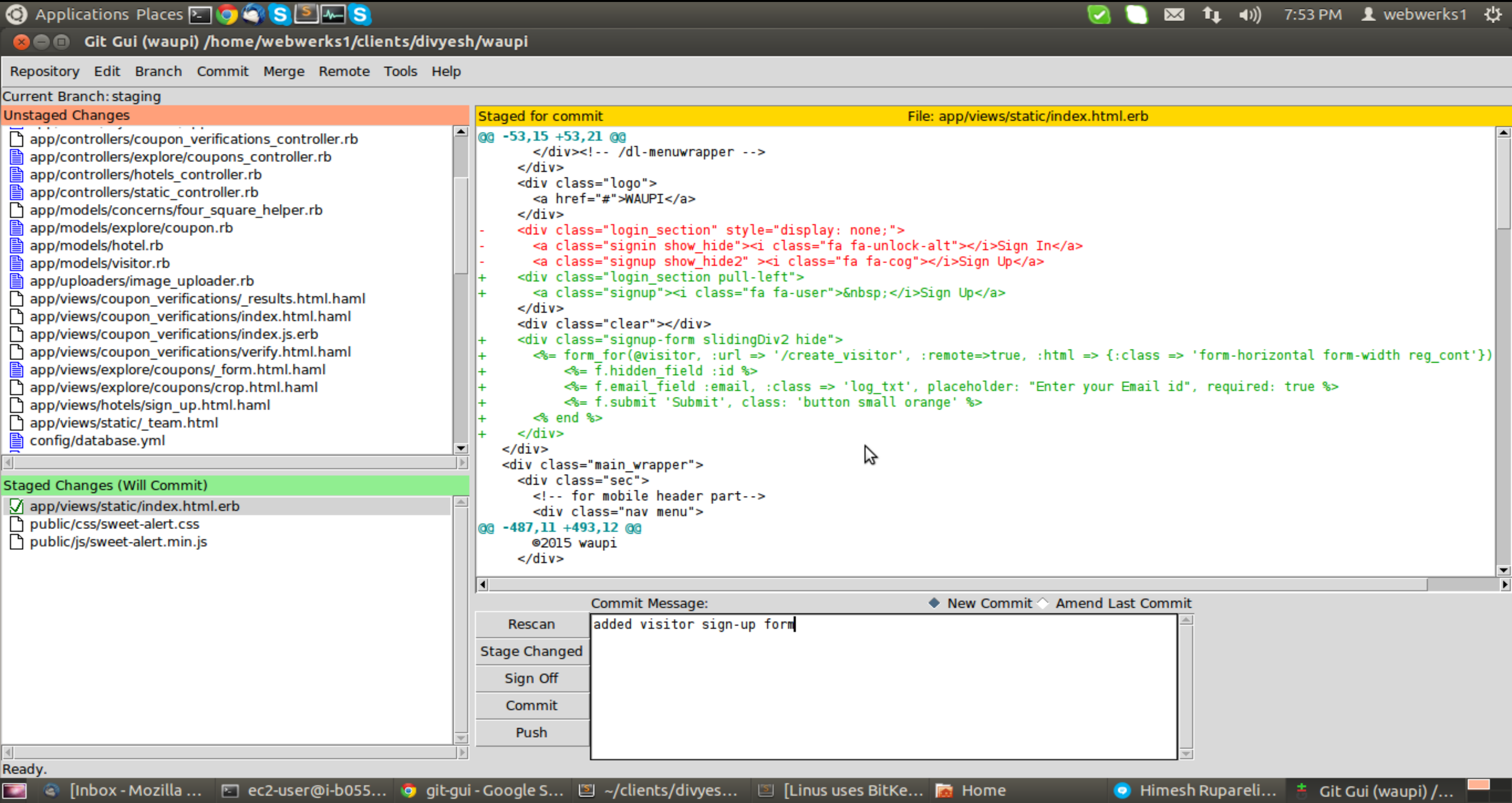
Release 0.3.1

628f7795bdd0f43c7769ec33c38b3ebc6c39879c

- 19 NEWS

```
@@ -1,3 +1,22 @@
+ = gitg 0.3.1 =
+ Version 0.3.1 was released on 2013-02-01
+
+ This is the first release of a major rewrite of gitg. We have since moved from
+ C to Vala to implement gitg, easing development. More importantly, gitg is now
+ using libgit2 (instead of calling out to git) which vastly improves gitg's
+ quality. The interface has also had a significant overhaul, conforming to the
+ latest GNOME 3 apps interface development guidelines.
+
+ Although the most prominent features of gitg are present in this release, there
+ are still some notable regressions which will be implemented in subsequent
+ releases and this release should therefore be considered unstable.
+
+ == Known regressions ==
+   * Push, pull branch
```

Git-GUI (git gui)



Current Branch: staging

Unstaged Changes

- app/controllers/coupon_verifications_controller.rb
- app/controllers/explore/coupons_controller.rb
- app/controllers/hotels_controller.rb
- app/controllers/static_controller.rb
- app/models/concerns/four_square_helper.rb
- app/models/explore/coupon.rb
- app/models/hotel.rb
- app/models/visitor.rb
- app/uploaders/image_uploader.rb
- app/views/coupon_verifications/_results.html.haml
- app/views/coupon_verifications/index.html.haml
- app/views/coupon_verifications/index.js.erb
- app/views/coupon_verifications/verify.html.haml
- app/views/explore/coupons/_form.html.haml
- app/views/explore/coupons/crop.html.haml
- app/views/hotels/sign_up.html.haml
- app/views/static/_team.html
- config/database.yml

Staged Changes (Will Commit)

- ☑ app/views/static/index.html.erb
- public/css/sweet-alert.css
- public/js/sweet-alert.min.js

Staged for commit File: app/views/static/index.html.erb

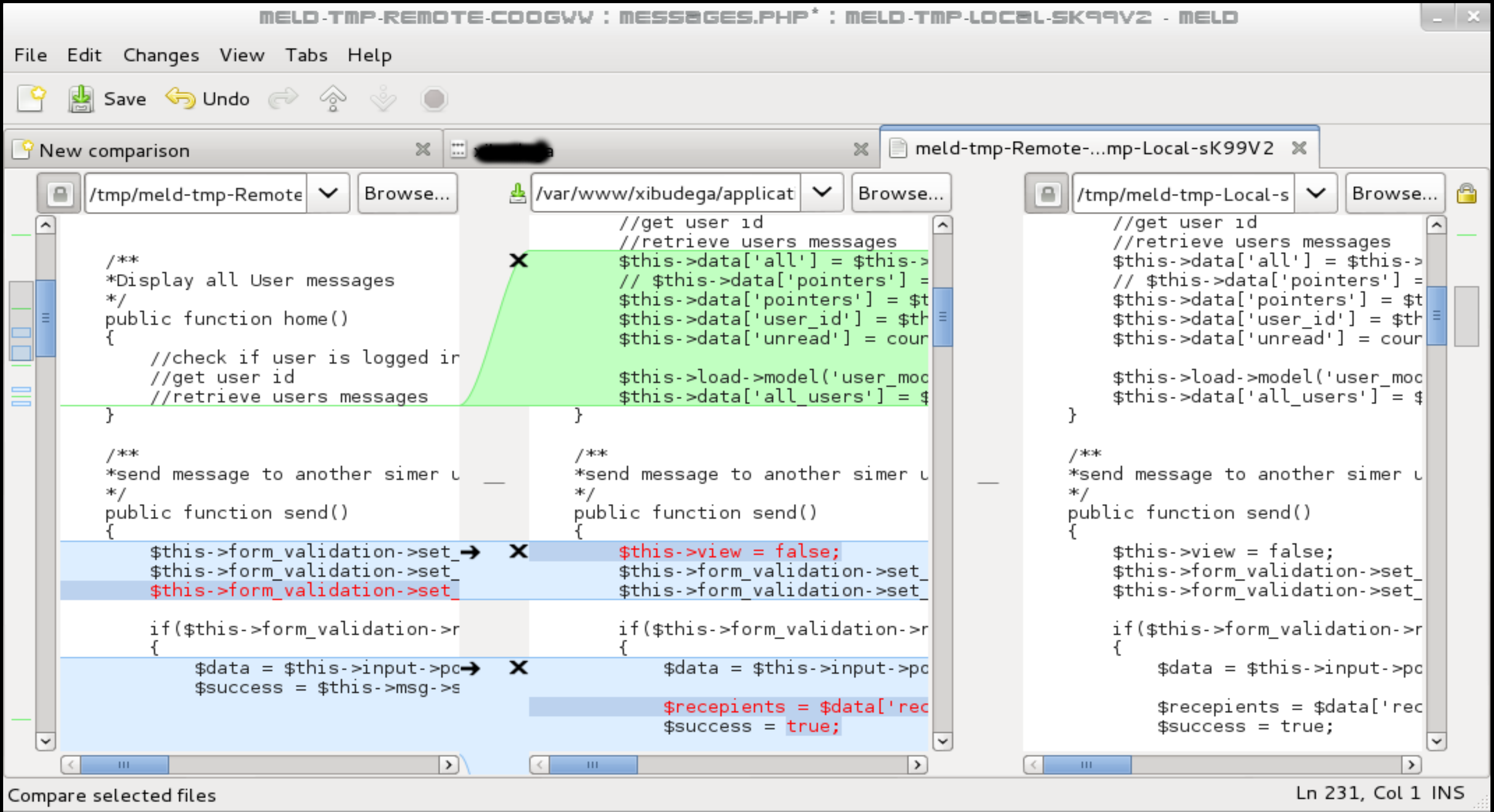
```
@@ -53,15 +53,21 @@
     </div><!-- /dl-menuwrapper -->
    </div>
    <div class="logo">
     <a href="#">WAUPI</a>
    </div>
-   <div class="login_section" style="display: none;">
-     <a class="signin show_hide"><i class="fa fa-unlock-alt"></i>Sign In</a>
-     <a class="signup show_hide2" ><i class="fa fa-cog"></i>Sign Up</a>
+   <div class="login_section pull-left">
+     <a class="signup"><i class="fa fa-user">&nbsp;</i>Sign Up</a>
    </div>
    <div class="clear"></div>
+   <div class="signup-form slidingDiv2 hide">
+     <%= form_for(@visitor, :url => '/create_visitor', :remote=>true, :html => {:class => 'form-horizontal form-width reg_cont'}) do |f| %>
+       <%= f.hidden_field :id %>
+       <%= f.email_field :email, :class => 'log_txt', placeholder: "Enter your Email id", required: true %>
+       <%= f.submit 'Submit', class: 'button small orange' %>
+     <% end %>
+   </div>
  </div>
  <div class="main_wrapper">
    <div class="sec">
     <!-- for mobile header part-->
     <div class="nav menu">
@@ -487,11 +493,12 @@
     ©2015 waupi
    </div>
```

Commit Message: ● New Commit ○ Amend Last Commit

```
added visitor sign-up form
```

Ready.

Meld (git mergetool)



```
    /**
    *Display all User messages
    */
    public function home()
    {
        //check if user is logged ir
        //get user id
        //retrieve users messages
    }

    /**
    *send message to another simer u
    */
    public function send()
    {
        $this->form_validation->set_
        $this->form_validation->set_
        $this->form_validation->set_

        if($this->form_validation->r
        {
            $data = $this->input->pc
            $success = $this->msg->s
```

```
        //get user id
        //retrieve users messages
        $this->data['all'] = $this->
        // $this->data['pointers'] =
        $this->data['pointers'] = $t
        $this->data['user_id'] = $th
        $this->data['unread'] = cour

        $this->load->model('user_moc
        $this->data['all_users'] = $
    }

    /**
    *send message to another simer u
    */
    public function send()
    {
        $this->view = false;
        $this->form_validation->set_
        $this->form_validation->set_

        if($this->form_validation->r
        {
            $data = $this->input->pc

            $recepients = $data['rec
            $success = true;
```

```
        //get user id
        //retrieve users messages
        $this->data['all'] = $this->
        // $this->data['pointers'] =
        $this->data['pointers'] = $t
        $this->data['user_id'] = $th
        $this->data['unread'] = cour

        $this->load->model('user_moc
        $this->data['all_users'] = $
    }

    /**
    *send message to another simer u
    */
    public function send()
    {
        $this->view = false;
        $this->form_validation->set_
        $this->form_validation->set_

        if($this->form_validation->r
        {
            $data = $this->input->pc

            $recepients = $data['rec
            $success = true;
```

Thank You