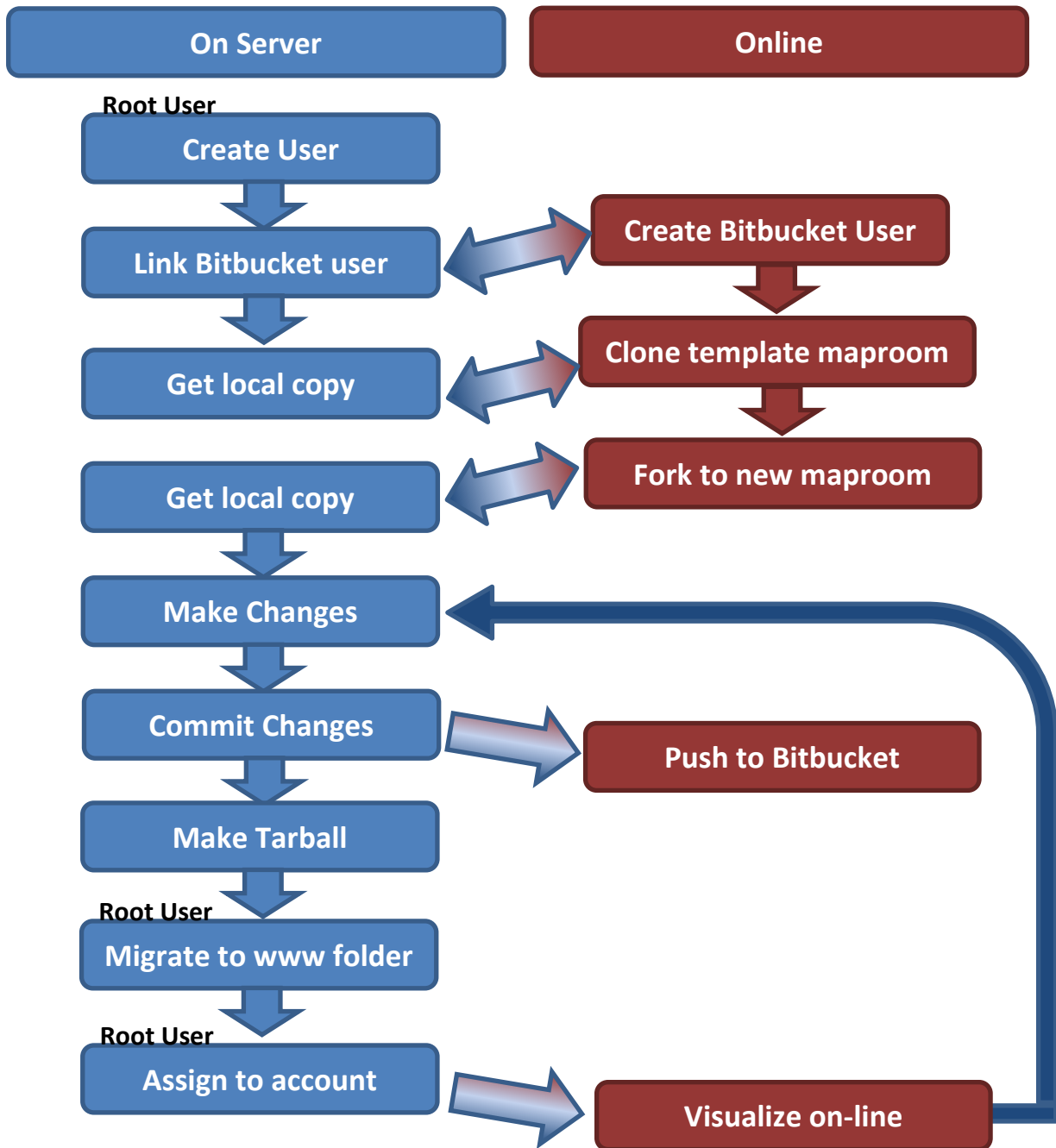


Manual for Chilean Data Library Management – Part 1

Contents

The maproom development process	2
Download PuTTY software for interaction with the server.....	3
Create new user in the Data Library.....	5
Create new user in Bitbucket or receive invitation	6
Setup Bitbucket under your account.....	7
Introduce yourself to Git	7
Generate and setup certificate	7
Setup command prompt (optional)	10
Setup git aliases (optional)	11
Cloning a local copy of a maproom to your account.....	12
Fork a new Maproom off of the original maproom_template_chile.....	14
Make changes to the master, commit and push to the central repository	17
Migrate the maproom to an online location.....	20
How to add new maproom ‘accounts’	23
Making a new tag for the maproom	24
Make a new branch ‘cazalac’ based on master.....	24

The maproom development process



Download PuTTY software for interaction with the server

Go to <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

PuTTY Download Page

[Home](#) | [Licence](#) | [FAQ](#) | [Docs](#) | [Download](#) | [Known Issues](#)
[Mirrors](#) | [Updates](#) | [Feedback](#) | [Changes](#) | [Wishlist](#)

Here are the PuTTY files themselves:

- PuTTY (the Telnet and SSH client itself)
- PSCP (an SCP client, i.e. command-line secure file copy)
- PSFTP (an SFTP client, i.e. general file transfer sessions much like FTP)
- PuTTYtel (a Telnet-only client)
- Plink (a command-line interface to the PuTTY back ends)
- Pageant (an SSH authentication agent for PuTTY, PSCP, PSFTP, and Plink)
- PuTTYgen (an RSA and DSA key generation utility).

LEGAL WARNING: Use of PuTTY, PSCP, PSFTP and Plink is illegal in countries where encryption is outlawed. I believe it is legal to use Pageant, but you should seek legal advice before downloading it. You may find [this site](#) useful (it's a survey of cryptography laws).

Use of the Telnet-only binary (PuTTYtel) is unrestricted by any cryptography laws.

There are cryptographic signatures available for all the files we offer below. We also supply cryptographically signed lists of checksums. To use a Windows program to compute MD5 checksums, you could try the one at [this site](#). (This MD5 program is also cryptographically signed by me.)

Binaries

The latest release version (beta 0.63). This will generally be a version I think is reasonably likely to work well. If you have a problem with it, please report it to me, but please make sure you have already fixed the bug, before reporting it to me.

For Windows on Intel x86

PuTTY:	putty.exe	(or by FTP)	(RSA sig)	(DSA sig)
PuTTYtel:	puttytel.exe	(or by FTP)	(RSA sig)	(DSA sig)
PSCP:	pscp.exe	(or by FTP)	(RSA sig)	(DSA sig)
PSFTP:	psftp.exe	(or by FTP)	(RSA sig)	(DSA sig)
Plink:	plink.exe	(or by FTP)	(RSA sig)	(DSA sig)
Pageant:	pageant.exe	(or by FTP)	(RSA sig)	(DSA sig)
PuTTYgen:	puttygen.exe	(or by FTP)	(RSA sig)	(DSA sig)

Open the PuTTY program and create a new session with these characteristics:

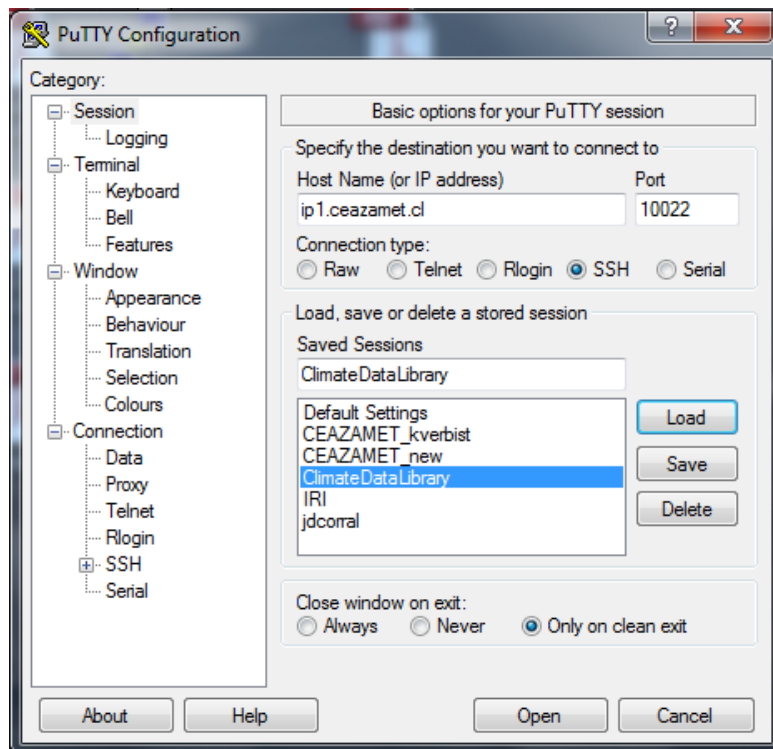
Host name: ip1.ceazamet.cl

Port: 10022

Connection type: SSH

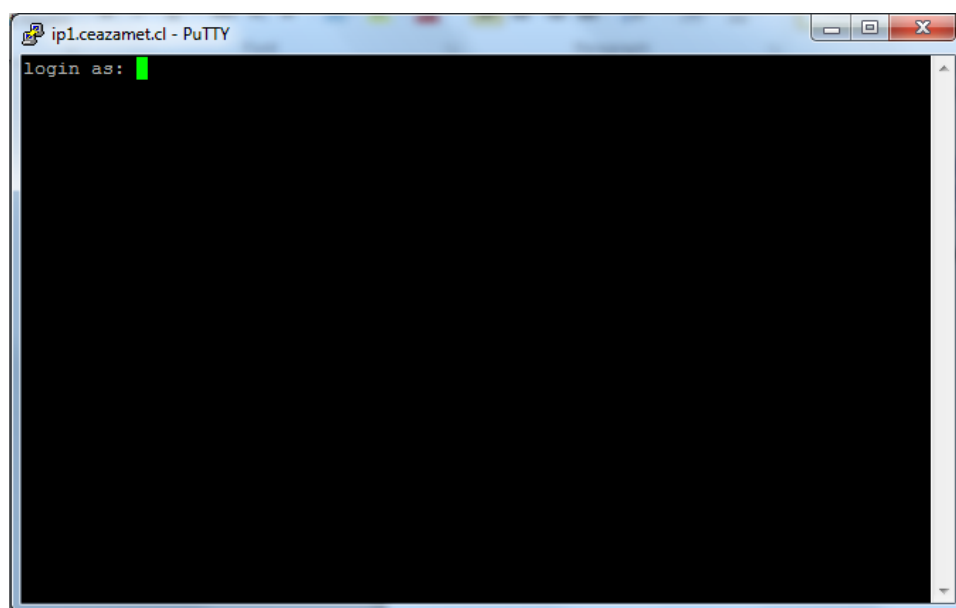
Saved Session: ClimateDataLibrary

Then click 'Save'

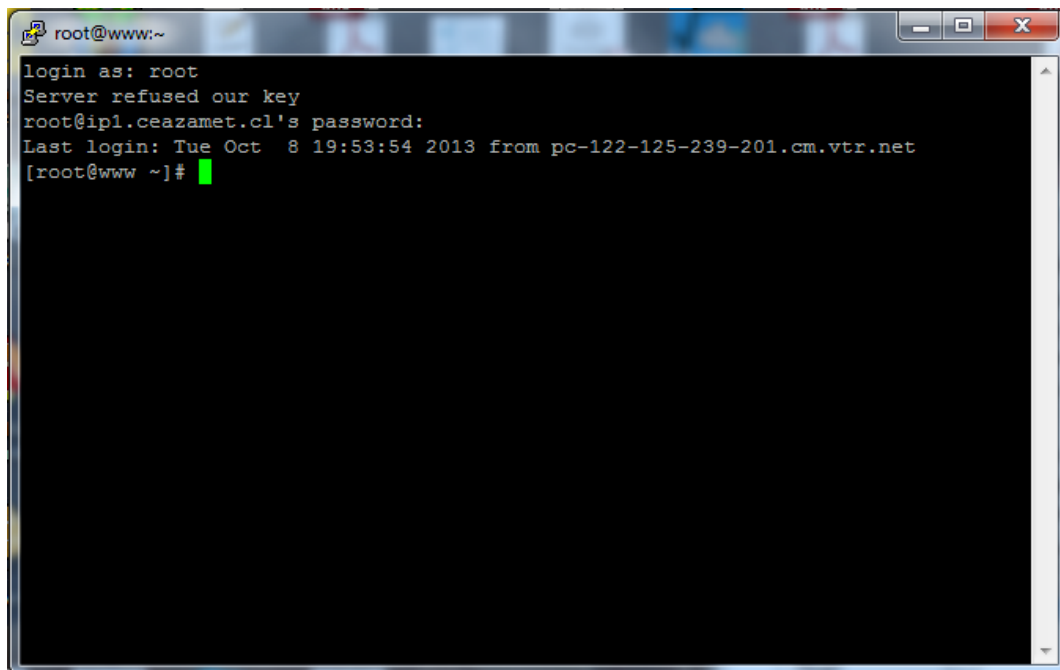


Then click 'Open' or double click on the recently created session.

When login appears: type 'root' (login as: root)



Type in the root password. You are now logged in as a root user.

A terminal window titled 'root@www:~' with standard window controls. The terminal output shows a login attempt as root, a message 'Server refused our key', a password prompt, and a successful login. The last login details are: 'Last login: Tue Oct 8 19:53:54 2013 from pc-122-125-239-201.cm.vtr.net'. The prompt is '[root@www ~]#' with a green cursor.

```
root@www:~  
login as: root  
Server refused our key  
root@ipl.ceazamet.cl's password:  
Last login: Tue Oct 8 19:53:54 2013 from pc-122-125-239-201.cm.vtr.net  
[root@www ~]#
```

Create new user in the Data Library

After entering as root user to the data library, you can create your user account:

To add a user:

```
>useradd kverbist2
```

To give it a password:

```
>passwd kverbist2
```

If you want to eliminate a user:

```
>userdel kverbist2
```

The creation of a user implicates that automatically the folder `/home/user_name` will be created, that will be assigned to this user

Create new user in Bitbucket or receive invitation

You can create a new user for the maproom code repository 'Bitbucket', entering directly into the website: <https://bitbucket.org/>

Or you can request an invitation from a user, which is the preferred option.

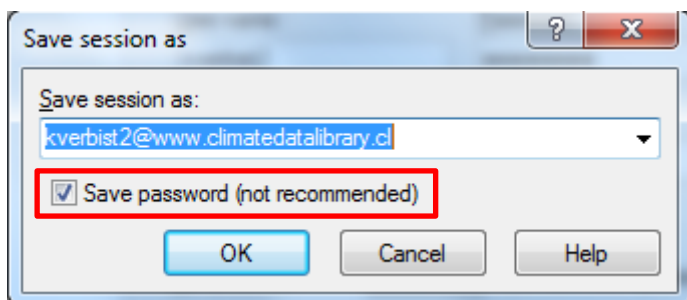
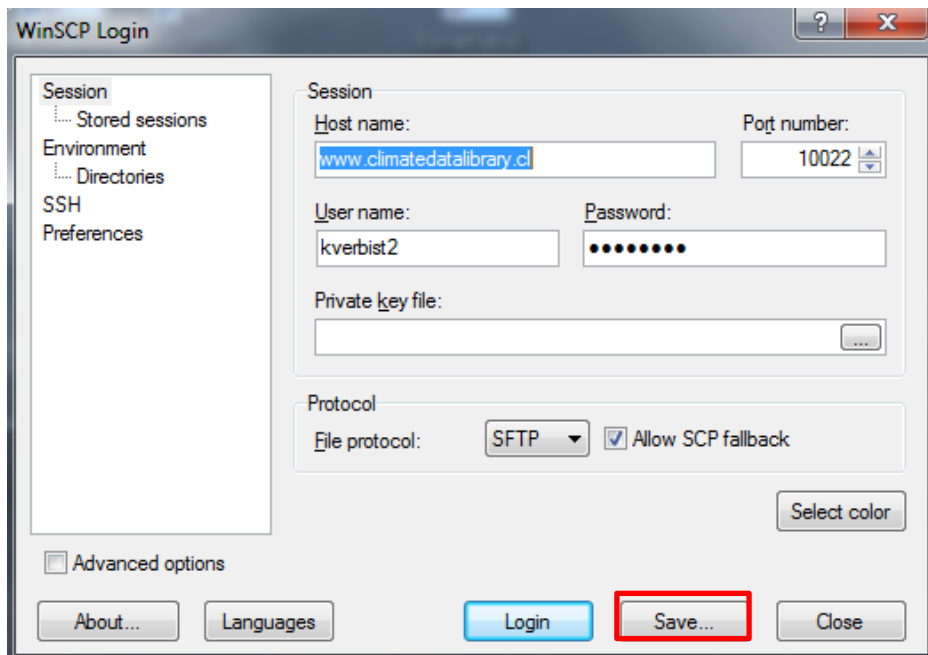
The screenshot shows the Bitbucket web interface. At the top is a dark blue header with the Bitbucket logo, a 'Repositories' dropdown, a 'Create' button, a search bar containing 'owner/repository', and user profile icons. Below the header is a yellow notification bar stating: 'It's time for kverbist to upgrade to the next user level — committers currently have read-only access. Upgrade or manage users.'

The main content area is titled 'Dashboard' and includes a 'Create repository' button. Below this is a tabbed interface with 'Overview' selected, 'Pull requests', and 'Issues'. The 'Overview' tab displays a commit history on the left, featuring three entries by 'Igor Khomyakov' dated '2013-09-27' and '2013-09-26'. Each entry lists merged commits and their descriptions, such as 'Merged iridl/maproom_template into master' and 'introduced RULESET option, now you may build maproom using differ...'. On the right side of the dashboard, there is a 'Send invitation' button highlighted with a red rectangle, accompanied by a progress bar showing 5 out of 8 invitations and icons for email and user profiles. Below this is a 'Repositories' section with a 'Create a repository' link. It includes a 'Filter repositories' input field and tabs for 'All', 'Watching', 'Mine', and 'Teams'. A list of repositories follows, including 'kverbist / maproom_cazalac', 'kverbist / maproom_cazalac2', 'kverbist / maproom_template_Chile', and 'kverbist / maproom_unea'.

```
[root@www kverbist2]# cat <<eos >> ~/.ssh/config
> Host bitbucket.org
> HostName bitbucket.org
> IdentityFile ~/.ssh/bitbucket
> eos
[root@www kverbist2]# chmod go-rwx ~/.ssh/config
```

4. Create a session in WinSCP with your new user

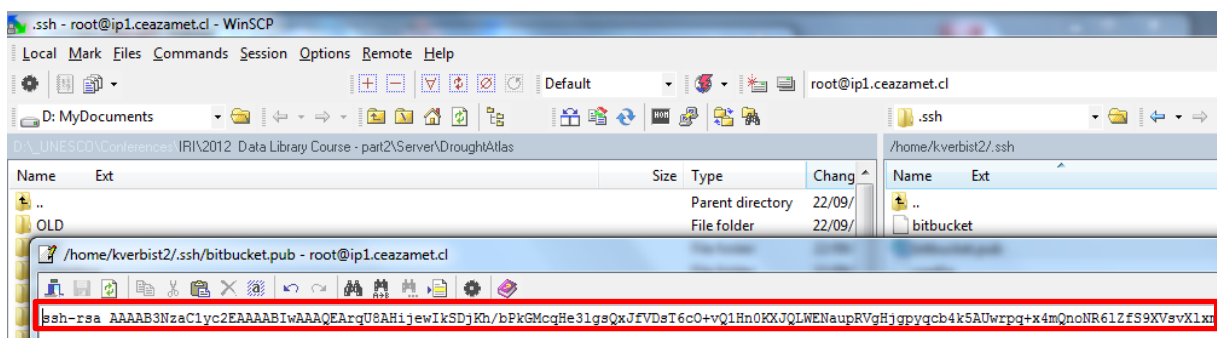
How to download and setup WinScp is explained in the previous manual and can be found [here](#).



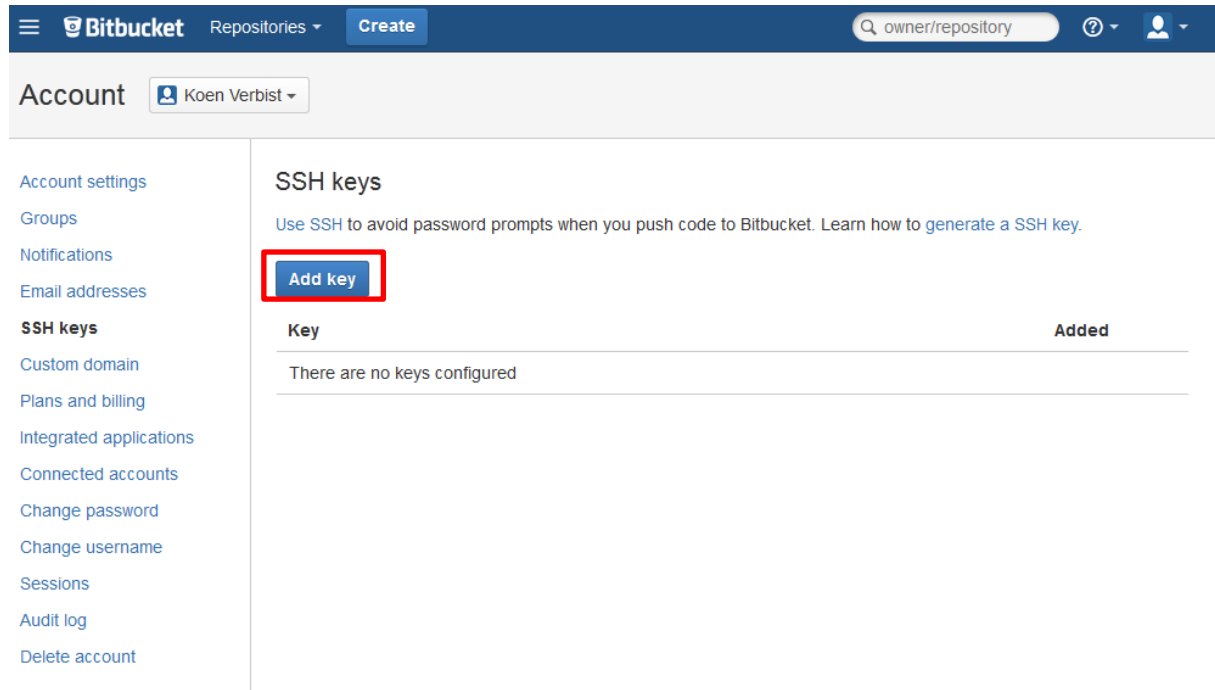
Now login this session as your new user.

5. Add your certificate to [Bitbucket](#) as follows:

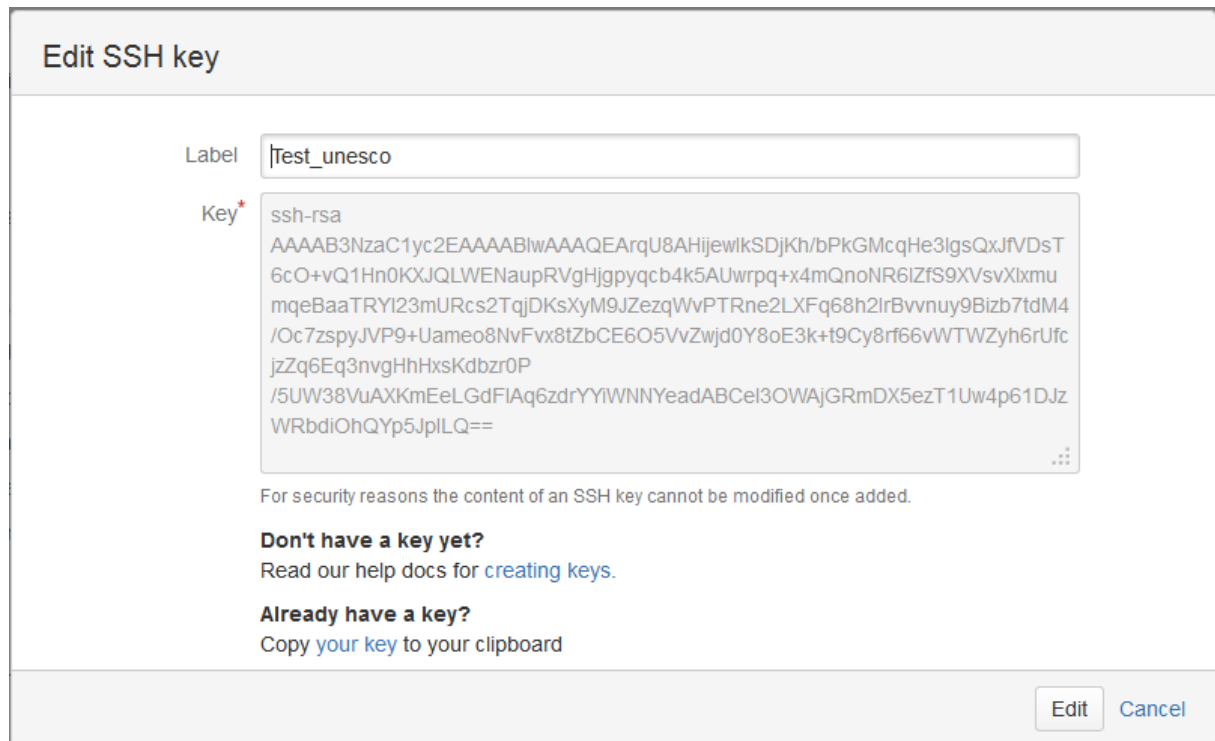
- Open the file bitbucket.pub that is located under /home/yourname/.ssh/
- Copy the contents of ~/.ssh/bitbucket.pub to SSH Key field. Make sure that you copy the text of the public key exactly without extra spaces or carriage returns.



- Go to the location in bitbucket that holds all SSH keys: [Bitbucket](#) SSH location
- Set `Label` field to some unique value that identifies the key, e.g. `work`, `home`
- Paste the contents of `~/.ssh/bitbucket.pub` to `SSH Key` field. Make sure that you paste the text of the public key exactly without extra spaces or carriage returns.
- Press `Add key`



The screenshot shows the Bitbucket web interface. At the top, there's a navigation bar with the Bitbucket logo, 'Repositories' dropdown, a 'Create' button, a search bar containing 'owner/repository', and user avatars. Below this is the 'Account' section for 'Koen Verbist'. On the left is a sidebar menu with links: Account settings, Groups, Notifications, Email addresses, SSH keys (highlighted), Custom domain, Plans and billing, Integrated applications, Connected accounts, Change password, Change username, Sessions, Audit log, and Delete account. The main content area is titled 'SSH keys' and includes a link to learn how to generate a SSH key. A red box highlights the 'Add key' button. Below it is a table with columns 'Key' and 'Added', currently showing 'There are no keys configured'.



The screenshot shows the 'Edit SSH key' dialog box. It has a 'Label' field with the text 'Test_unesco' and a 'Key*' field containing a long SSH public key string. Below the key field, there's a note: 'For security reasons the content of an SSH key cannot be modified once added.' There are two sections: 'Don't have a key yet?' with a link to 'creating keys', and 'Already have a key?' with a link to 'Copy your key to your clipboard'. At the bottom right are 'Edit' and 'Cancel' buttons.

The screenshot shows the Bitbucket web interface. At the top, there's a navigation bar with the Bitbucket logo, 'Repositories' dropdown, a 'Create' button, a search bar with 'owner/repository', and user icons. Below this is the 'Account' section for 'Koen Verbist'. A sidebar on the left lists account settings: Account settings, Groups, Notifications, Email addresses, SSH keys (highlighted), Custom domain, Plans and billing, Integrated applications, Connected accounts, and Change password. The main content area is titled 'SSH keys' and includes a link to learn how to generate an SSH key and an 'Add key' button. Below this is a table of SSH keys:

Key	Added
Test_unesco	55 seconds ago Edit x

5. Run a test to check if connection can be made:

```
>ssh -T -i ~/.ssh/bitbucket git@bitbucket.org
```

If successful, you should see something like this:

You can use git or hg to connect to Bitbucket.

```
[kverbist2@www ~]$ ssh -T -i ~/.ssh/bitbucket git@bitbucket.org
The authenticity of host 'bitbucket.org (131.103.20.168)' can't be established.
RSA key fingerprint is 97:8c:1b:f2:6f:14:6b:5c:3b:ec:aa:46:46:74:7c:40.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added 'bitbucket.org,131.103.20.168' (RSA) to the list of known hosts.
logged in as kverbist2.
You can use git or hg to connect to Bitbucket. Shell access is disabled.
```

Setup command prompt (optional)

You may set up your command prompt to show which branch you are on. Add the following line to your ~/.bashrc:

```
function parse_git_branch () {
    git branch 2> /dev/null | sed -e '/^[^*]/d' -e 's/* \(.*)/ (\1)/'
}
RED="\[\033[0;31m\]"
YELLOW="\[\033[0;33m\]"
GREEN="\[\033[0;32m\]"
NO_COLOUR="\[\033[0m\]"
PS1="$YELLOW$h:\w$RED\$(parse_git_branch)$NO_COLOUR\$ "

alias g=git

export PATH=$HOME/bin:/usr/local/miconf/bin:/usr/local/semantic_tools/bin:$PATH
```

```
/home/kverbist/.bashrc - kverbist@www.climatedatalibrary.cl

# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific aliases and functions

function parse_git_branch () {
    git branch 2> /dev/null | sed -e '/^\(^*\)/d' -e 's/* \(.*/ (\1)/'
}
RED="\[\033[0;31m\]"
YELLOW="\[\033[0;33m\]"
GREEN="\[\033[0;32m\]"
NO_COLOUR="\[\033[0m\]"
PS1="$YELLOW\h:\w$RED\$ (parse_git_branch) $NO_COLOUR\$ "

alias g=git

export PATH=$HOME/bin:/usr/local/miconf/bin:/usr/local/semantic_tools/bin:$PATH
```

This will display, the following prompt:

www:~/maproom_template_chile (master)\$

```
www:~$ cd maproom_template_chile/
www:~/maproom_template_chile (master)$
```

Setup git aliases (optional)

Copt paste the following code into the terminal:

```
cat <<eos >> ~/.gitconfig
```

```
[alias]
l = log --graph --pretty=format:'%Cred%h%Creset -%C(yellow)%d%Creset %s %Cgreen(%cr)
%C(bold blue)<an>%Creset' --abbrev-commit --date=relative
s = status
a = add
d = diff --color
b = branch
ci = commit
co = checkout
sb = show-branch
ds = diff --cached
f = reflog
r = remote -v
sm = submodule
smu = submodule update --init --recursive
lgg = log --all --graph --decorate
lgr = log --all --graph --decorate --oneline --simplify-by-decoration --no-merges
lgt = log --format='%h %an %ar - %s'
lgl = log --oneline origin..HEAD
lgf = log --oneline HEAD..origin/master
h = help
    cl = clone --recursive
cfd = clean -fd
cffd = clean -ffd
eos
```

Cloning a local copy of a maproom to your account

As a first test, clone the template maproom:

1. Go to the bitbucket page of this maproom: https://bitbucket.org/kverbist/maproom_template_chile
And copy the SSH link

The screenshot shows the Bitbucket web interface for the repository 'maproom_template_Chile' owned by 'kverbist'. The repository is a fork of 'maproom_template'. The page includes a header with the Bitbucket logo, a search bar, and navigation tabs for Overview, Source, Commits, Pull requests, Issues, Wiki, and Downloads. Below the header, there's a section for building the maproom, listing required software: Java 1.6.0_10+, GNU Make 3.81+, Unix shell, Raptor 1.4.18+, and semantic_tools 1.2.0+. To the right, the SSH link is displayed as 'git@bitbucket.org:kverbist/mapro'. Below the SSH link, there are statistics for 2 Branches, 3 Tags, 0 Forks, and 1 Watcher. A table on the right provides details about the repository: Owner (koen verbist), Access level (Private), Type (Git), Last updated (20 hours ago), Created (2013-09-25), and Size (728.5 KB). The 'Recent activity' section shows a push by Igor Khomyakov 20 hours ago, with two commits: 'd7aa56f - Merged iridl/maproom_template into master' and 'dfc229e - updated maproomtools; fixes Imports/maproom_bulletins.html error'.

To build maproom you need the following software:

- Java 1.6.0_10+
- GNU Make 3.81+
- Unix shell
- Raptor 1.4.18+
- semantic_tools 1.2.0+

Recent activity

Igor Khomyakov pushed 2 commits to kverbist/maproom_template_Chile 20 hours ago

- d7aa56f - Merged iridl/maproom_template into master
- dfc229e - updated maproomtools; fixes Imports/maproom_bulletins.html error

SSH `git@bitbucket.org:kverbist/mapro`

2 Branches 3 Tags 0 Forks 1 Watcher

Owner	koen verbist
Access level	Private
Type	Git
Last updated	20 hours ago
Created	2013-09-25
Size	728.5 KB (download)

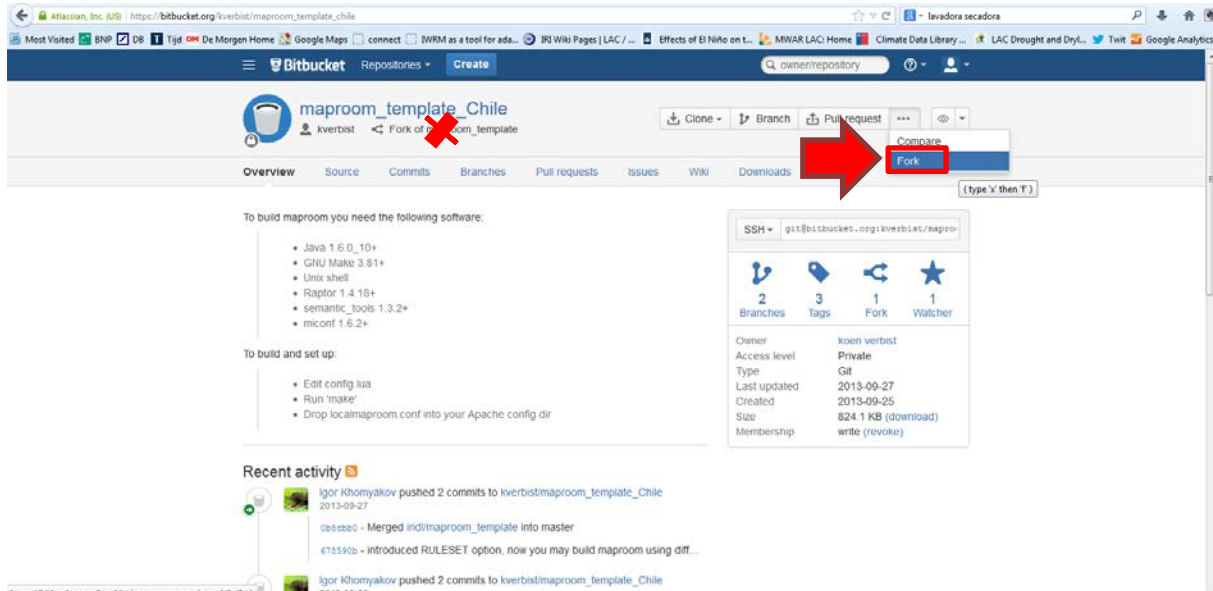
2. Go to the terminal and clone the repository recursively (including subfolders)
>git clone --recursive git@bitbucket.org:kverbist/maproom_template_chile.git

```
[kverbist2@www ~]$ git clone --recursive git@bitbucket.org:kverbist/maproom_template_chile.git
Initialized empty Git repository in /home/kverbist2/maproom_template_chile/.git/
remote: Counting objects: 53, done.
remote: Compressing objects: 100% (51/51), done.
remote: Total 53 (delta 21), reused 0 (delta 0)
Receiving objects: 100% (53/53), 28.48 KiB, done.
Resolving deltas: 100% (21/21), done.
Submodule 'maproomtools' (git@bitbucket.org:iridl/maproomtools.git) registered for path 'maproomtools'
Submodule 'miconf' (git@bitbucket.org:iridl/miconf.git) registered for path 'miconf'
Submodule 'pure' (git@bitbucket.org:iridl/pure.git) registered for path 'pure'
Submodule 'uicore' (git@bitbucket.org:iridl/uicore.git) registered for path 'uicore'
Initialized empty Git repository in /home/kverbist2/maproom_template_chile/maproomtools/.git/
remote: Counting objects: 295, done.
remote: Compressing objects: 100% (200/200), done.
remote: Total 295 (delta 178), reused 163 (delta 94)
Receiving objects: 100% (295/295), 46.70 KiB, done.
Resolving deltas: 100% (178/178), done.
Submodule path 'maproomtools': checked out '1470cb81385de144f1a71708646658464815b50b'
Initialized empty Git repository in /home/kverbist2/maproom_template_chile/miconf/.git/
remote: Counting objects: 352, done.
remote: Compressing objects: 100% (302/302), done.
remote: Total 352 (delta 150), reused 171 (delta 42)
Receiving objects: 100% (352/352), 302.39 KiB | 544 KiB/s, done.
Resolving deltas: 100% (150/150), done.
Submodule path 'miconf': checked out 'e68eb47b50cbd9a538ede4862845af47c646ebcd'
Submodule 'lua' (git@bitbucket.org:iridl/lua.git) registered for path 'lua'
Initialized empty Git repository in /home/kverbist2/maproom_template_chile/miconf/lua/.git/
remote: Counting objects: 295, done.
remote: Compressing objects: 100% (174/174), done.
remote: Total 295 (delta 118), reused 295 (delta 118)
Receiving objects: 100% (295/295), 502.38 KiB | 61 KiB/s, done.
Resolving deltas: 100% (118/118), done.
Submodule path 'lua': checked out 'a1e97d785a742f8d2d22ac48bea3bee589cd5d63'
Initialized empty Git repository in /home/kverbist2/maproom_template_chile/pure/.git/
remote: Counting objects: 60, done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 60 (delta 16), reused 0 (delta 0)
Receiving objects: 100% (60/60), 381.65 KiB | 42 KiB/s, done.
Resolving deltas: 100% (16/16), done.
Submodule path 'pure': checked out 'd63b8bad39a9dafc68103d76d2495828d16716b9'
Initialized empty Git repository in /home/kverbist2/maproom_template_chile/uicore/.git/
remote: Counting objects: 792, done.
remote: Compressing objects: 100% (570/570), done.
remote: Total 792 (delta 395), reused 472 (delta 222)
Receiving objects: 100% (792/792), 246.08 KiB | 232 KiB/s, done.
Resolving deltas: 100% (395/395), done.
Submodule path 'uicore': checked out '2742bef8fe5dd1f6a4b80b979f71623f4bdb5f76'
```

Fork a new Maproom off of the original maproom_template_chile

Go to bitbucket and login

Click 'Fork' (NOT Fork of maproom_template)



Give a name and description and press 'Fork Repository'

Fork kverbist / maproom_template_Chile

Name*

Description

It's encouraged to write a little about why you are forking.

Access level ☒ This is a private repository
This repository does not allow public forks.

Forking

Permissions ☒ Inherit repository user/group permissions

Project management ☒ Issue tracking
☒ Wiki



What is a fork?

Forking is a great way to contribute to a project even though you don't have write access. Check out our documentation for more details.

Go to administration:

Bitbucket Repositories **Create** ?

maproom_test
 kverbist2 Fork of maproom_template_Chile
 Share

Clone Branch Pull request  

Overview Source Commits Branches Pull requests Issues Wiki Downloads

To build maproom you need the following software:

- Java 1.6.0_10+
- GNU Make 3.81+
- Unix shell
- Raptor 1.4.18+
- semantic_tools 1.3.2+
- miconf 1.6.2+

To build and set up:

- Edit config.lua
- Run 'make'
- Drop localmaproom.conf into your Apache config dir


SSH

2 Branches 3 Tags 0 Forks 1 Watcher

Owner	Koen Verbist
Access level	Private
Type	Git
Last updated	just now
Created	just now
Size	767.2 KB (download)

In User Management, you can set user permissions and add/delete users

maproom_test
 kverbist2 Fork of maproom_template_Chile
 Share

Clone Branch Pull request ... 

Overview Source Commits Branches Pull requests Issues Wiki Downloads

Repository details

Access management

Branch management

Username aliases

Hooks

Links





Deployment keys

Transfer repository

Delete repository

Access management

Users

 Koen Verbist	owner			
 Igor Khomyakov		READ	WRITE	ADMIN
 koen verbist		READ	WRITE	ADMIN
 Rémi COUSIN		READ	WRITE	ADMIN

Now select the SSH-key to the maproom and copy (Select and click 'ctrl+C')

maproom_test
kverbist2 Fork of maproom_template_Chile

Clone Branch Pull request

Overview Source Commits Branches Pull requests Issues Wiki Downloads

To build maproom you need the following software:

- Java 1.6.0_10+
- GNU Make 3.81+
- Unix shell
- Raptor 1.4.18+
- semantic_tools 1.3.2+
- miconf 1.6.2+

To build and set up:

- Edit config.lua
- Run 'make'
- Drop localmaproom.conf into your Apache config dir

SSH `git@bitbucket.org:kverbist2/maproom_test.git`

2 Branches 3 Tags 0 Forks 1 Watcher

Owner	Koen Verbist
Access level	Private
Type	Git
Last updated	a minute ago
Created	a minute ago
Size	767.2 KB (download)

Go to the terminal and go to your home directory:

Now clone the repository by typing 'git clone --recursive' and paste the new repository ssh-key (click 'ctrl+V')

```
>git clone --recursive git@bitbucket.org:kverbist2/maproom_test.git
```

```
www:~$ git clone --recursive git@bitbucket.org:kverbist2/maproom_test.git
Initialized empty Git repository in /home/kverbist2/maproom_test/.git/
Warning: Permanently added the RSA host key for IP address '131.103.20.167' to the list of known hosts.
remote: Counting objects: 75, done.
remote: Compressing objects: 100% (73/73), done.
remote: Total 75 (delta 33), reused 0 (delta 0)
Receiving objects: 100% (75/75), 31.70 KiB, done.
Resolving deltas: 100% (33/33), done.
Submodule 'maproomtools' (git@bitbucket.org:iridl/maproomtools.git) registered for path 'maproomtools'
Submodule 'pure' (git@bitbucket.org:iridl/pure.git) registered for path 'pure'
Submodule 'uicore' (git@bitbucket.org:iridl/uicore.git) registered for path 'uicore'
Initialized empty Git repository in /home/kverbist2/maproom_test/maproomtools/.git/
remote: Counting objects: 310, done.
remote: Compressing objects: 100% (215/215), done.
remote: Total 310 (delta 190), reused 163 (delta 94)
Receiving objects: 100% (310/310), 48.68 KiB, done.
Resolving deltas: 100% (190/190), done.
```

Now go to the folder of the maproom and update the submodules using the command, to make sure the scripts that create the maproom are updated to the latest version:

```
>cd maproom_test
```


>git submodule update --init --recursive

```
www:~$ ls
git  maproom_cazalac  maproom_template_chile  maproom_test
www:~$ cd maproom_test
www:~/maproom_test (master)$ git submodule update --init --recursive
www:~/maproom_test (master)$
```

You now have a personal copy of the template maproom, which you can modify and expand as wanted, without affecting any other users.

Make changes to the master, commit and push to the central repository

As a quick test to see if all is setup well, we make 1 small change to the maproom.

Go to the maproom section within maproom_test

>cd maproom

Check available files:

>ls

```
www:~/maproom_test (master)$ cd maproom
www:~/maproom_test/maproom (master)$ ls
Examples  index.xhtml.en
```

We see 1 xhtml file and 1 folder that holds additional xhtml and html files. As a test, we make a change to index.xhtml.en

Open the file in the WinSCP environment (**make sure you are logged in as your user, and NOT as root**).

Change the title >CEAZA map room< to >kverbist2 map room<

```
/home/kverbist2/maproom_test/maproom/index.xhtml.en - kverbist2@www.climatedatalibrary.cl
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://iridl.ldeo.columbia.edu/ontolog
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:wms="http://www.opengis.net/wms#"
      xmlns:iridl="http://iridl.ldeo.columbia.edu/ontologies/iridl.owl#"
      xmlns:maproom="http://iridl.ldeo.columbia.edu/ontologies/maproom.owl#"
      xmlns:term="http://iridl.ldeo.columbia.edu/ontologies/iriterms.owl#"
      xmlns:xs="http://www.w3.org/2001/XMLSchema#"
      version="XHTML+RDFa 1.0"
      xml:lang="en"
    >
<head>
<meta name="viewport" content="width=device-width; initial-scale=1.0;" />
<title>CEAZA Map Room</title>
<link rel="stylesheet" type="text/css" href="../uicore/uicore.css" />
<link class="altLanguage" rel="alternate" hreflang="es" href="index.html?Set-Language=es" />
<link class="altLanguage" rel="alternate" hreflang="fr" href="index.html?Set-Language=fr" />
<link rel="canonical" href="index.html" />
<link rel="term:isDescribedBy" href="http://iridl.ldeo.columbia.edu/ontologies/iridl.owl#climate" />
<link rel="term:isDescribedBy" href="http://iridl.ldeo.columbia.edu/ontologies/iridl.owl#forecast" />
<script type="text/javascript" src="../uicore/uicore.js"></script>
</head>
<body>

<form name="pageform" id="pageform">
<input class="carryLanguage" name="Set-Language" type="hidden" />
<input class="titleLink itemImage" name="bbox" type="hidden" />
</form>
<div class="controlBar">
    <fieldset class="navitem">
        <legend>CEAZA</legend>
        <a rev="section" class="navlink carryup" href="http://iridl.ldeo.columbia.edu/">
    </fieldset>
    <fieldset class="navitem">
        <legend>Data Library</legend>
        <span class="navtext">Maproom</span>
    </fieldset>
</div>
```

Now go to the terminal and check the status of the git repository:

>git status

```
www:~/maproom_test (master)$ git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   maproom/index.xhtml.en
#
no changes added to commit (use "git add" and/or "git commit -a")
```

A file has been modified from its original.

Now add the file to the files you want to add to the online repository 'Bitbucket'.

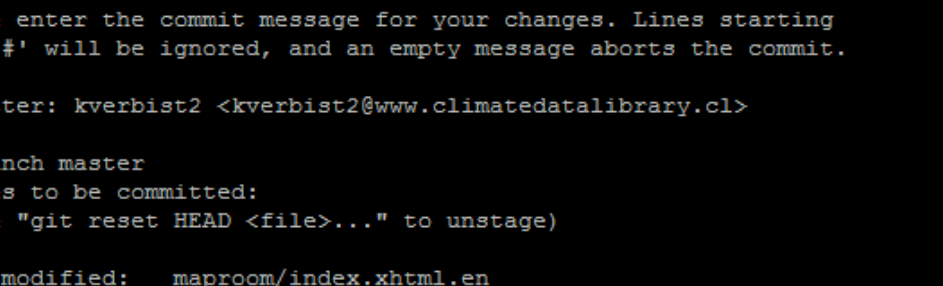
>git add *

And commit the changes to the file sent to the repository

>git commit -a

```
www:~/maproom_test (master)$ git add *
www:~/maproom_test (master)$ git commit -a
```

A new screen will appear that requests to describe the changes made (comment).

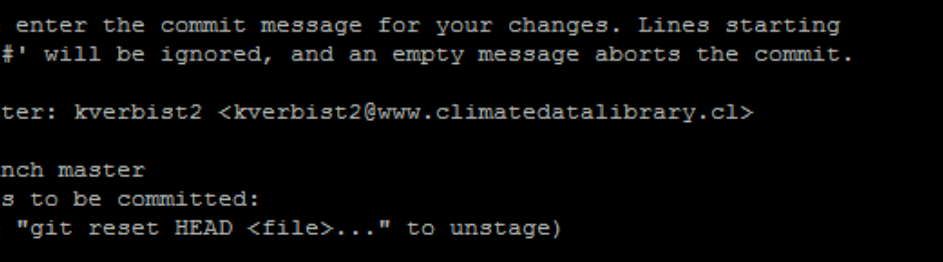


```
kverbist2@www:~/maproom_test

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Committer: kverbist2 <kverbist2@www.climatedatalibrary.cl>
#
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   maproom/index.xhtml.en
#
~
~
~
~
~
~
~
~
~
~
~

".git/COMMIT_EDITMSG" 12L, 336C
```

Now enter i to start inserting text, the bottom changes to -- INSERT--



```
kverbist2@www:~/maproom_test

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Committer: kverbist2 <kverbist2@www.climatedatalibrary.cl>
#
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   maproom/index.xhtml.en
#
~
~
~
~
~
~
~
~
~
-- INSERT --
```

Now type the changes made: "Changed title to kverbist2 map room"

If finished, press ESC and type

:wq

to exit the editor (wq stands for **w**rite and **q**uit). You will return to the prompt and see following message:

```
www:~/maproom_test (master)$ git commit -a
[master a6c1483] Changed title to kverbist2 map room
Committer: kverbist2 <kverbist2@www.climatedatalibrary.cl>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

If the identity used for this commit is wrong, you can fix it with:

    git commit --amend --author='Your Name <you@example.com>'

1 files changed, 1 insertions(+), 1 deletions(-)
```

To check you are well connected to the repository, you can push your commitment to the repo

>git push -u origin master

```
www:~/maproom_test (master)$ git push -u origin master
Counting objects: 7, done.
Delta compression using up to 6 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 409 bytes, done.
Total 4 (delta 2), reused 0 (delta 0)
To git@bitbucket.org:kverbist2/maproom_test.git
    0b8cbb0..a6c1483  master -> master
Branch master set up to track remote branch master from origin.
```

From now on, you can use git push to send commitments.

>git push

```
www:~/maproom_test (master)$ git push
Everything up-to-date
```

Migrate the maproom to an online location

Make sure you are in the main folder of the maproom \maproom_test

>pwd

```
www:~/maproom_test (master)$ pwd
/home/kverbist2/maproom_test
```

Now create a tarball (compressed version of the maproom):

>make tarball

```
www:~/maproom_test (master)$ make tarball
git-generate-version-info maproom xml >maproom/version.xml
cd maproom; ../maproomtools/build_maproom.pl owl-max-optimized;
Using ruleset 'owl-max-optimized'
Building maproomregistry.
mar oct  8 23:07:13 CLT 2013
Setting '/home/kverbist2/maproom_test/maproom/newmaproomcache' directory
Gathering rdfa triples
```

IMPORTANT: open up a new console and login as root!!

Then go to /var/www/html/

>cd /var/www/html/

Untar (de-inflate) the maproom in this folder (**your version will vary**)

> tar xvfz ~kverbist2/maproom_test/maproom-1.0.2-20-7a3f9551e3e89f04ee4808aa30103b.tgz

```
[root@www maproom_test]# cd /var/www/html/
[root@www html]# tar xvfz ~kverbist2/maproom_test/maproom-1.0.2-20-ga6c148369d7a
3f9551e3e89f04ee4808aa30103b.tgz
```

Now remove the current link to the maproom

>rm -f kverbist2

Now create a new virtual link to the maproom with freshly created maproom

(ln -s maproom_name link_name)

>ln -s maproom-1.0.2-20-7a3f9551e3e89f04ee4808aa30103b kverbist2

```
[root@www html]# ln -s maproom-1.0.2-20-ga6c148369d7a3f9551e3e89f04ee4808aa30103
b/ kverbist2
```

Now go to the online version of the maproom:

<http://www.climatedatalibrary.cl/kverbist2/maproom/>

You can see the title of the page reads now: **kverbist2 map room**

kverbist2 Map Room - Mozilla Firefox

File Edit View History Bookmarks Tools Help

MWAR LAC: News x Sitio Privado de Clientes - Banchile In... x New Tab x kver

www.climatedatalibrary.cl/kverbist2/maproom/

Most Visited BNP DB Tijd DM De Morgen Home Google Maps connect IWRM as a tool for ada... IRI Wiki Pages | L

CEAZA

Climate Data Library

Data Library

Maproom

Region

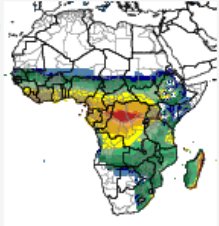
Global

CEAZA Map Room







The maproom is a collection of maps and other figures that monitor climate and societal conditions at present and in the recent past. The maps and figures can be manipulated and are linked to the original data. Even if you are primarily interested in data rather than figures, this is a good place to see which datasets are particularly useful for monitoring current conditions.

Examples

Climate affects sectors in society in a number of ways. These effects may be direct, as with heat stress, or indirect, as with infectious diseases such as malaria and meningitis.



Share

 0  +1 Recommend this on Google

Congratulations: you implemented your first maproom!

How to add new maproom 'accounts'

First go to /usr/local/squid/etc

```
login as: root
Server refused our key
root@ipl.ceazamet.cl's password:
Last login: Tue Aug 20 18:31:20 2013 from vpn158.ugent.be
[root@www ~]# cd /usr/local/squid/etc
[root@www etc]#
```

First make a backup of the current (working) config file, using the current date:

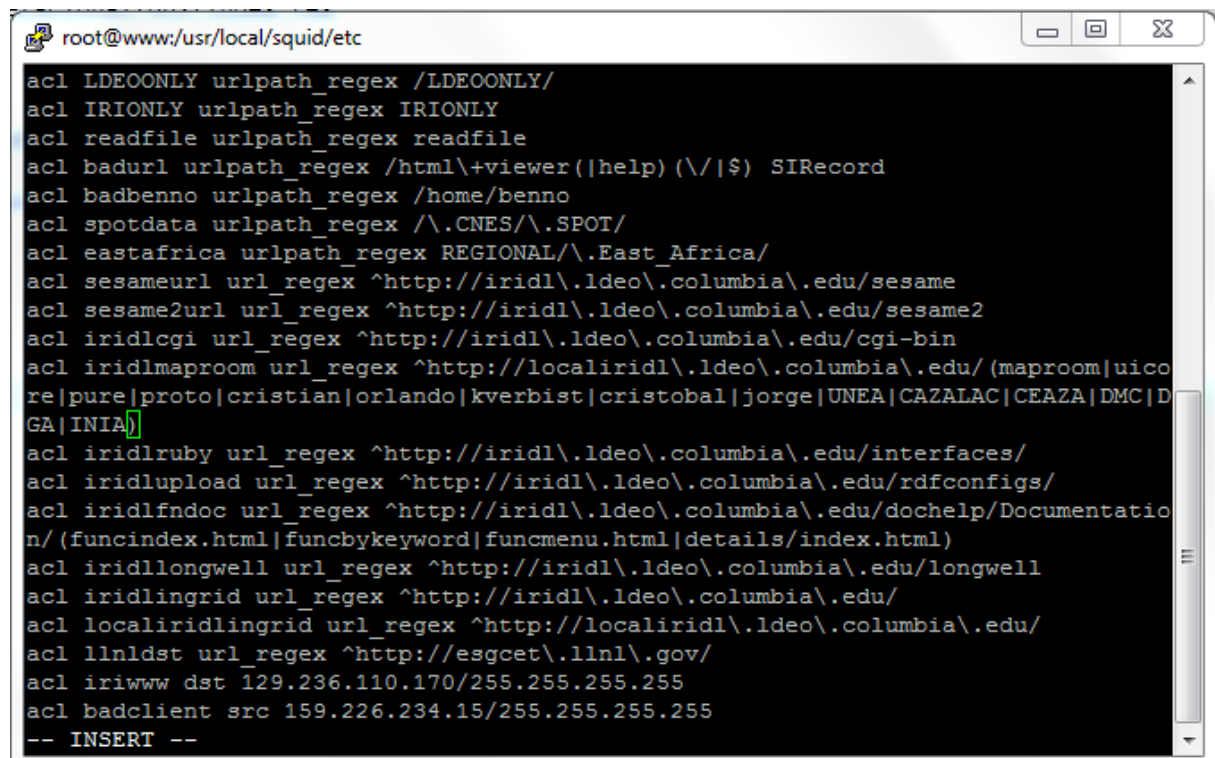
```
>cp squid.conf squid.conf.20130517
```

Now edit the config file using

```
>vi squid.conf
```

And search for the line that states 'acl iridlmaproom url_regex ^http://localiridl...'

You will find a list of already defined maproom names, such as UNEA and CAZALAC, to which you can add additional names, after using a pipe symbol '|'.



```
root@www:/usr/local/squid/etc
acl LDEOONLY urlpath_regex /LDEOONLY/
acl IRIONLY urlpath_regex IRIONLY
acl readfile urlpath_regex readfile
acl badurl urlpath_regex /html\+viewer(|help)(\|/|$) SRecord
acl badbenno urlpath_regex /home/benno
acl spotdata urlpath_regex /\.CNES/\..SPOT/
acl eastafrica urlpath_regex REGIONAL/\..East_Africa/
acl sesameurl url_regex ^http://iridl\..ldeo\..columbia\..edu/sesame
acl sesame2url url_regex ^http://iridl\..ldeo\..columbia\..edu/sesame2
acl iridlcgi url_regex ^http://iridl\..ldeo\..columbia\..edu/cgi-bin
acl iridlmaproom url_regex ^http://localiridl\..ldeo\..columbia\..edu/(maproom|uico
re|pure|proto|cristian|orlando|kverbist|cristobal|jorge|UNEA|CAZALAC|CEAZA|DMC|D
GA|INIA)
acl iridlruby url_regex ^http://iridl\..ldeo\..columbia\..edu/interfaces/
acl iridupload url_regex ^http://iridl\..ldeo\..columbia\..edu/rdfconfigs/
acl iridlndoc url_regex ^http://iridl\..ldeo\..columbia\..edu/dochelp/Documentatio
n/(funcindex.html|funcbykeyword|funcmenu.html|details/index.html)
acl iridllongwell url_regex ^http://iridl\..ldeo\..columbia\..edu/longwell
acl iridlingrid url_regex ^http://iridl\..ldeo\..columbia\..edu/
acl localiridlingrid url_regex ^http://localiridl\..ldeo\..columbia\..edu/
acl llndst url_regex ^http://esgcet\..llnl\..gov/
acl iriwww dst 129.236.110.170/255.255.255.255
acl badclient src 159.226.234.15/255.255.255.255
-- INSERT --
```

In order for the changes to come into effect, the squid needs to be reconfigured:

```
>../sbin/squid -k parse
```

```
>../sbin/squid -k reconfigure
```

```
[root@www etc]# ../sbin/squid -k parse
[root@www etc]# ../sbin/squid -k reconfigure
```

Now stop and restart ingrid and squid

```
>sudo /etc/init.d/ingrid stop
>sudo /etc/init.d/squid stop
>sudo /etc/init.d/squid start
>sudo /etc/init.d/ingrid start
```

```
[root@www etc]# sudo /etc/init.d/ingrid stop
```

```
[root@www etc]# sudo /etc/init.d/squid stop
Parando squid: ..... [ OK ]
```

```
[root@www etc]# sudo /etc/init.d/squid start
Iniciando squid: [ OK ]
[root@www etc]# sudo /etc/init.d/ingrid start
```

In order to check if Ingrid is running satisfactorily, you should check the service is running:

```
>ps -ef | grep ingrid
```

```
[root@www etc]# ps -ef | grep ingrid
root      27173      1  0 21:48 pts/0    00:00:00 ./ingridd
root      27185 26871  0 21:49 pts/0    00:00:00 grep ingrid
```

Making a new tag for the maproom

First create a new tag

```
>git tag maproom_xyz-1.0.0
```

```
>git push --tags
```

Then adjust the Makefile with the following changes

```
VER = $(shell miconf/scripts/git-generate-version-info maproom_xyz tag)
```

```
VER_ID = $(shell miconf/scripts/git-generate-version-info maproom_xyz id)
```

Make a new branch 'cazalac' based on master

```
>git checkout -b cazalac master
```

Now you can make changes to the local version of the master under this new branch. If you're finished:

```
>git status
```

```
>git add *
```

```
>git commit -a
```

Push the new branch and its first commit to the central repository (called 'origin'):

>git push -u origin cazalac

```
www:~/maproom_cazalac (cazalac)$ git push -u origin cazalac
Counting objects: 19, done.
Delta compression using up to 6 threads.
Compressing objects: 100% (12/12), done.
Writing objects: 100% (12/12), 3.94 KiB, done.
Total 12 (delta 9), reused 0 (delta 0)
To git@bitbucket.org:kverbist/maproom_cazalac.git
 * [new branch]      cazalac -> cazalac
Branch cazalac set up to track remote branch cazalac from origin.
```

Now create a tarball (compressed version of the maproom):

>make tarball

```
www:~/maproom_cazalac (cazalac)$ make tarball
miconf/scripts/git-generate-version-info maproom xml >maproom/version.xml
cd maproom; \
    ../maproomtools/build_maproom.pl;
Building maproomregistry.
Gathering rdfa triples
jue ago 1 18:26:45 CLT 2013
In /home/kverbist/maproom_cazalac/maproom
```

IMPORTANT: open up a new console and login as root

Then go to /var/www/html/

>cd /var/www/html/

Untar (de-inflate) the maproom in this folder (**your version will vary**)

>tar xvfz ~kverbist/maproom_cazalac/maproom-1.3.0-19-ga1525f7ea2d8b9260fa4f54b0987e671cb

```
[root@www html]# tar xvfz ~kverbist/maproom_cazalac/maproom-1.3.0-19-ga1525f7ea2d8b9260fa4f54b0987e671cb6b2061.tgz
```

Now remove the current link to the maproom

>rm -f MAPROOM_NAME

Now create a new virtual link to the maproom with freshly created maproom

(ln -s maproom_name link_name)

>ln -s maproom-1.3.0-19-ga1525f7ea2d8b9260fa4f54b0987e671cb MAPROOM_NAME

```
[root@www html]# ln -s maproom-1.3.0-19-ga1525f7ea2d8b9260fa4f54b0987e671cb6b2061/ CAZALAC
```