Vogsphere - Manual

Presidential form for turn in requests

zaz zaz@staff.42.fr

Summary: *This manual explains how to use the Vogsphere, a.k.a 42' turn-in system,*
including git basics. Git is the version-control software the Vogsphere uses.

Contents

Chapter I

Forewords

I.1 Git

- Git is a distributed version control manager. It's the tool we use to store and version your turn-ins for virtually every project at 42.

- It's your responsability to acquire knowledge of Git and its use. This manual only covers a very limited aspect of this incredible software. One can acquire this knowledge very fast by reading git's documentation online. This will save you lots of troubles.

I.2 A few directions

Here are a couple of links where you can find good quality documentations regarding Git:

- The official documentation: git-scm.com

- This amazing tool

Chapter II

Vogsphere operations

II.1 Summary

- When you register to a project, a unique and personnal repository for you of your team and for this specific project is automaticaly created.

- You can find your repository's url on the intranet on your project's page.

- A repository's url looks more or less like this :

```
vogsphere@vogsphere.42.fr:pisc-0-001/2013/paris-0-1/j00/yourlogin
```

- Next, you have to fetch a fresh local copy of your repository using the `git clone` command.

- Next, you're going to work on this local copy. Once done, or whenever you'd like to notify Git of your work, you must :

  - Add the modifications by using the `git add` command.

  - Create a new revision by using the `git commit` command.

- You can check the status of your local repository at anytime by using the `git status` command.

- By commiting your work step by step, you can create as much revisions as you want. Read the documentation to understand why it's a good idea to do so...

- To send your revisions to Vogsphere's side (i.e. to the repository that's going to be used for evaluations), you have to use the `git push` command. On your very first push to a repository, don't forget to specify the destination (`git push origin master`).

- After a successful push, all your revisions will exist on Vogsphere's copy of your repository and will be visible during evaluation.

- Obviously, any revisions created on your local copy after a push won't exist on Vogsphere's copy of your repository. You must push them as well, this is (fortunately) not automatic. It would be a shame to work hard on a project and forget to push it, wouldn't it ?

- If you wish to check what have been effectively turned in (i.e. which data is available on Vogsphere's copy of your repository) at anytime, you can clone a fresh local copy of your repository in a temporary folder. This clone will be the exact copy of what you would have turned in if the evaluation would happen right now.

- When you work as a team (for some projects or rushes), you will face synchronisation issues between each team member's local copy of your team's repository. Several solution to solve this exist and are covered in detail in Git's documentation...

II.2   Exemple

Here is a quick sample of operating a repository using Vogspere (obviously, the urls will differ):

```
zaz@blackjack ~ $ cd work
zaz@blackjack work $ git clone vogsphere@vogsphere.42.fr:pisc-0-001/2013/paris-0-1/d00/demo d00
Cloning into 'd00'...
vogsphere: (INFO) This transaction has been started at 2013-06-17 16:19:35, server time.
vogsphere: (INFO) Rights will be determined using this time, so do NOT cut the connection.
vogsphere: (INFO) You have read and write rights on this repository
warning: You appear to have cloned an empty repository.
zaz@blackjack work $ cd d00
zaz@blackjack d00 $ vim test.c
zaz@blackjack d00 $ git add test.c
zaz@blackjack d00 $ git commit -m "Added test.c"
[master (root-commit) be2069b] Added test.c
 1 file changed, 4 insertions(+)
 create mode 100644 test.c
zaz@blackjack d00 $ vim test2.c
zaz@blackjack d00 $ git add test2.c
zaz@blackjack d00 $ git commit -m "Added test2.c"
[master f195334] Added test2.c
 1 file changed, 6 insertions(+)
 create mode 100644 test2.c
zaz@blackjack d00 $ git push origin master
vogsphere: (INFO) This transaction has been started at 2013-06-17 16:22:52, server time.
vogsphere: (INFO) Rights will be determined using this time, so do NOT cut the connection.
vogsphere: (INFO) You have read and write rights on this repository
Counting objects: 9, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 897 bytes, done.
Total 9 (delta 1), reused 0 (delta 0)
To vogsphere@vogsphere.42.fr:pisc-0-001/2013/paris-0-1/d00/demo
 * [new branch]      master -> master
zaz@blackjack d00 $ echo "/* New modification */" >> test2.c
zaz@blackjack d00 $ git commit -am "Commiting a modification without git add, you fool"
[master ba0a82a] Commiting a modification without git add, you fool
 1 file changed, 1 insertion(+)
zaz@blackjack d00 $ git push
vogsphere: (INFO) This transaction has been started at 2013-06-17 16:23:35, server time.
vogsphere: (INFO) Rights will be determined using this time, so do NOT cut the connection.
vogsphere: (INFO) You have read and write rights on this repository
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 351 bytes, done.
Total 3 (delta 1), reused 0 (delta 0)
To vogsphere@vogsphere.42.fr:pisc-0-001/2013/paris-0-1/d00/demo
   a6f17c3..ba0a82a  master -> master
zaz@blackjack d00 $
```

II.3   Please remember the followings

> ⚠️ Vogsphere enforces a strict quota of 100MB per repository. Be very careful about that, because exceeding the quota will be considered as an abuse of resources, and will be punished as such !

- Vogsphere uses a Kerberos ticket to authentify you. A ticket is automaticaly acquired when you log in your session. However, after some time, this ticket will expire. If you are unable to use Vogsphere, you can acquire a new ticket by using the `kinit` command.

- Your turn-in repositories will close at the exact date and time specified on your project's page on the intranet. You will still have a read-access on your repository, but you won't be allowed to `push` data anymore.

- When you connect to Vogsphere, the date and time used to check your write access to your repository on Vogsphere is the date and time of your connection to Vogsphere. If too many students try to push data at the very same time, your request will be queued (Vogsphere will notify you). Don't worry, as long as you don't cut the connection or you make your push before the deadline, you push will be accepted if the turn in date and time expired. But anyway, who could be silly enough to turn his/her work in at the last momentcd...