Migration to git from Perforce

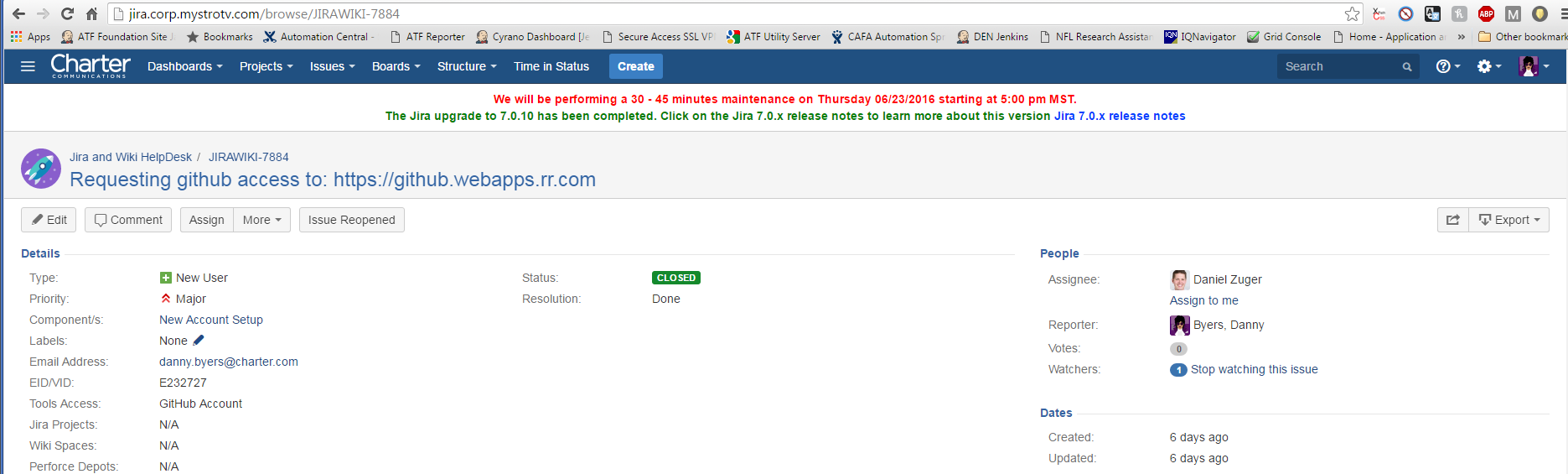
# Assumptions:

* Have a maven java project setup with the usual ATF classpath structure. Example: com.twc.sit.som
* Have git installed on your computer. See <https://github.webapps.rr.com/ATF/docs/tree/master/github_setup/git/install>
* Have github setup in your IDE. See either <https://github.webapps.rr.com/ATF/docs/tree/master/github_setup/eclipse-setup> or <https://github.webapps.rr.com/ATF/docs/tree/master/github_setup/intellij-setup> to the point where you are ready to 'Import project from GitHub' or 'Clone a Git Repository'. You will return to these instructions after the project is in github and you have created the README.md file.

# Download and install git

* go to <https://git-scm.com/downloads>
* click the download for you (mac or windows)
* install the dmg or msi
* open up a command prompt or terminal window
* type git -help
* if you get some help doc then everything is set up properly
* if not, you may need to add git to your $PATH variable
* On MAC, do this:
  + cd ~
  + vi .bashrc
  + press i to enter Insert mode
  + type in export $PATH=$PATH:/path/to/git/bin
  + press esc to exit Insert mode
  + type :wq!
  + now close the terminal
  + open another terminal and try git -help again
  + if that STILL doesn't work, then talk to Michael Wingfield, Ryan Yeash, or Jerry Dang
* On Windows, do:
  + open up Control Panel > System and Security > System
  + on the left side bar, select “Advanced System Settings”
  + a popup should appear, click the button named “Environment Variables”
  + on the section for your user, click the “PATH” variable
  + Under “System Variables”, find “PATH” and add the full directory path to github’s bin folder to the end of the “PATH” variable value (separated by semicolon ; )
  + open up a new command prompt and try “git –help” again

# Open jirawiki help desk ticket, [Requesting github access to: https://github.webapps.rr.com](http://jira.corp.mystrotv.com/browse/JIRAWIKI-7884)



* Once you have been given a github account, reach out to Danny Byers to set you up with git hub access to our projects.

# Update pom.xml and include repository info in your existing perforce projects (if not already there)

<repositories>

<repository>

  <id>sit-snapshots</id>

    <name>SITNexus Snapshots Repository</name>

    <url>http://mystro:mystro@atf-utility.corp.mystrotv.com:8081/nexus/content/repositories/snapshots/</url>

    <releases>

    <enabled>false</enabled>

    </releases>

    <snapshots>

    <enabled>true</enabled>

      <checksumPolicy>fail</checksumPolicy>

    </snapshots>

  </repository>

<repository>

  <id>sit-releases</id>

    <name>SITNexus Releases Repository</name>

    <url>http://mystro:mystro@atf-utility.corp.mystrotv.com:8081/nexus/content/repositories/releases/</url>

    <releases>

    <enabled>true</enabled>

    </releases>

    <snapshots>

    <enabled>false</enabled>

      <checksumPolicy>fail</checksumPolicy>

    </snapshots>

  </repository>

</repositories>

# Check in your last perforce changes, and then update/sync the perforce environment to make sure the latest code base is on your desktop workspace.

# In a command prompt window, or a git shell, change to the directory of the project that you wish to convert to git and perform the following commands (example shows AAD):

git init

git add .

git commit -m “some message”

git remote add origin <https://github.webapps.rr.com/CAFASIT/aad.git>

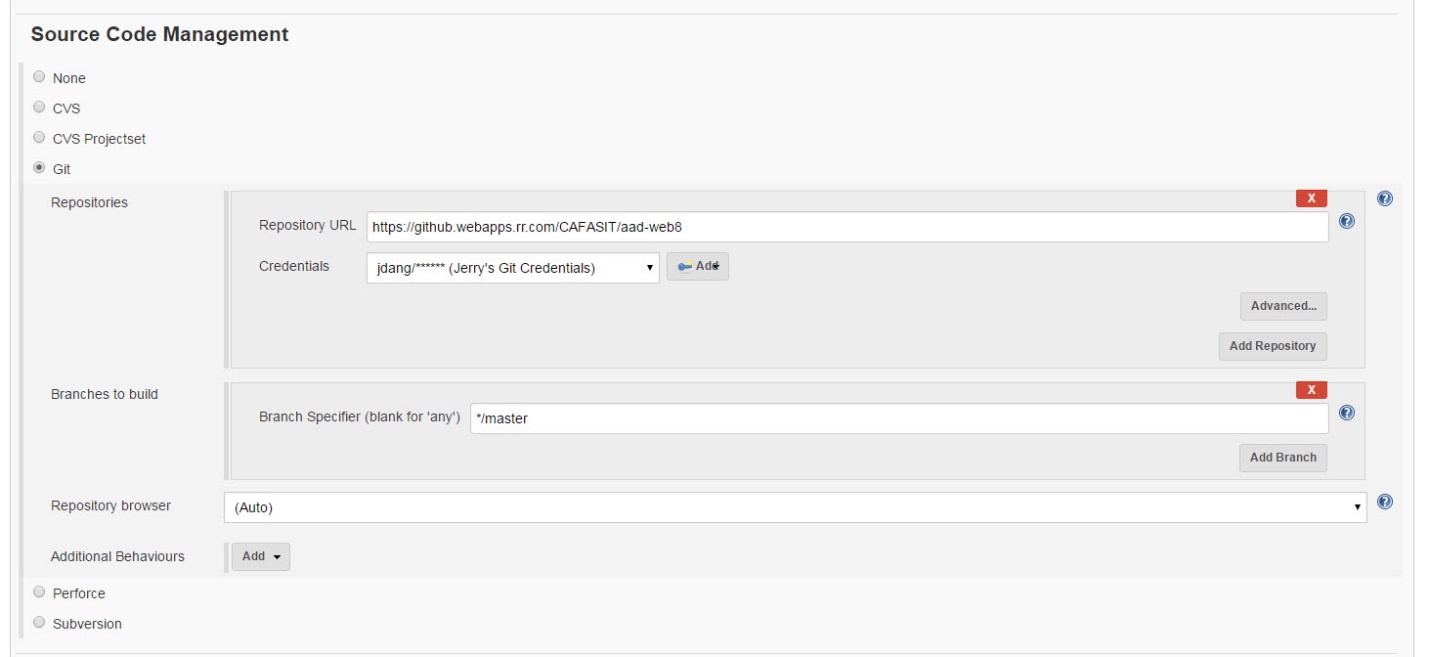
git pull origin master –allow-unrelated-histories

git push -u origin master

1. Verify that your code now shows up in github.
2. Configure Jenkins job to use github instead of perforce
   1. Add the GitBucket URL and project:



* 1. Change the Source Code Management from Perforce to Git and use your Github credentials to login. This is very important, as everyone will no longer have access to all projects to check in code.



# Troubleshooting

# removes the remote origin so git pull doesn’t do anything.

git remote rm origin

# how to push when readme is always there

git pull origin master --allow-unrelated-histories