Contents

[Preface 1](#_Toc442191144)

[SCM strategy and CI workflow 1](#_Toc442191145)

[1. Operate Git with TortoiseGit 2](#_Toc442191146)

[1.1 Generate SSH Keys 2](#_Toc442191147)

[1.2 Bind your SSH public key 5](#_Toc442191148)

[1.3 Clone GIT repositories 6](#_Toc442191149)

[1.4 Run junction to link three repo 6](#_Toc442191150)

[1.5 Migrate from old repo to current repo 7](#_Toc442191151)

[1.6 How to submit CI 7](#_Toc442191152)

[2. Operate Git with Command in console 7](#_Toc442191153)

[2.1 Generate SSH Keys and configure username 7](#_Toc442191154)

[2.2 Bind your SSH public key 9](#_Toc442191155)

[2.3 Clone GIT repositories 9](#_Toc442191156)

[2.4 Run junction to link three repo 9](#_Toc442191157)

[2.5 Migrate from old repo to current repo 10](#_Toc442191158)

[2.6 How to submit CI 10](#_Toc442191159)

# Preface

This document will not tell you how to install and configure Git or TortoiseGit client tool, if you have problem on it, pls refer to old document.

This document is mainly focus on illustrate you how to configure your environment to make sure you can clone from new repo and push your change to new repo hosted on Gitlab.

As SSH is deployed as GIT transport protocol. Due to the non-compatible to generate and use SSH key between GUI(TortoiseGit) and command. We will demonstrate you how to operate in both of these two ways one by one.

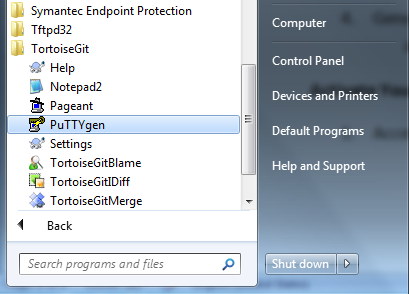
# SCM strategy and CI workflow

<https://docs.google.com/presentation/d/1D3_IYYBLT-eGKjW2PmMnT1Hmygmhs4Ymg0LsPu5dlZQ/edit#slide=id.gc6f980f91_0_0>

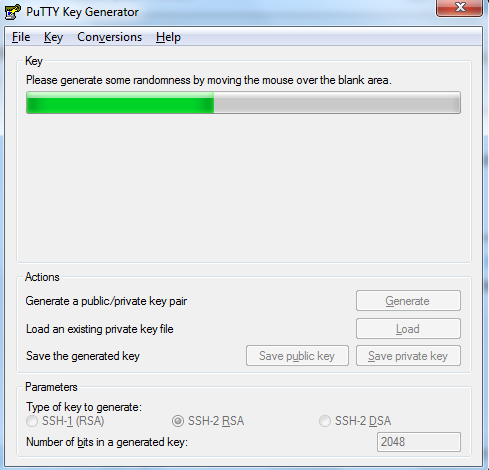
# 1. Operate Git with TortoiseGit

## 1.1 Generate SSH Keys

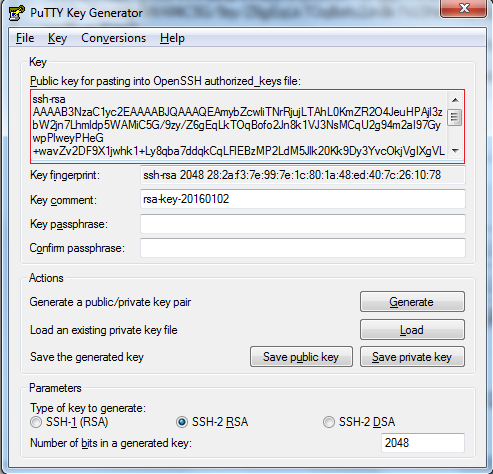
1. Run PuTTYgen:



1. Keep moving the mouse to generate SSH keys:



1. Click the save button to save your public and private key pair, the saved file will be used when you do git push, illustrate in section 1.6. And take care text in below red box, it is needed in section 1.2 to bind your SSH key to server.

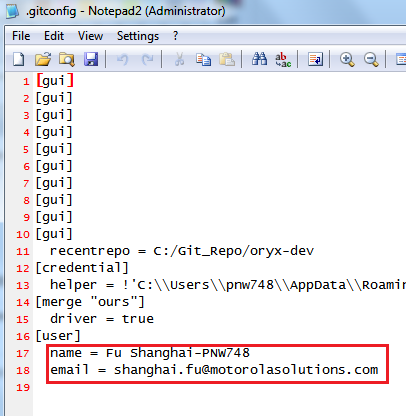
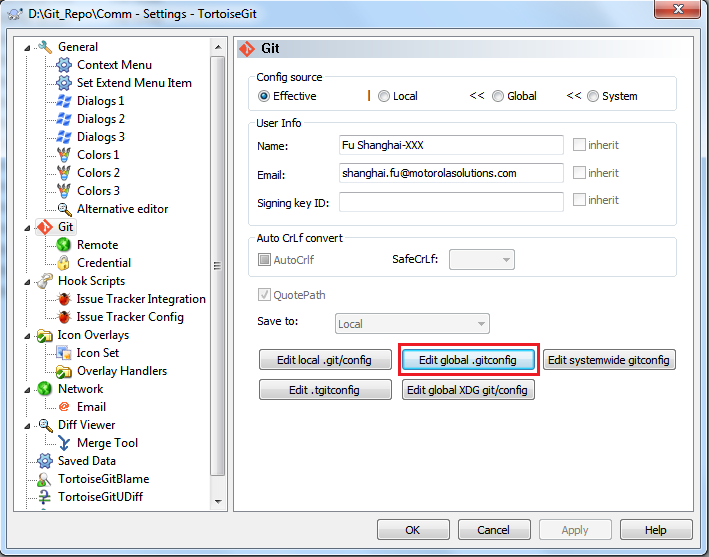


1. Configure username and email

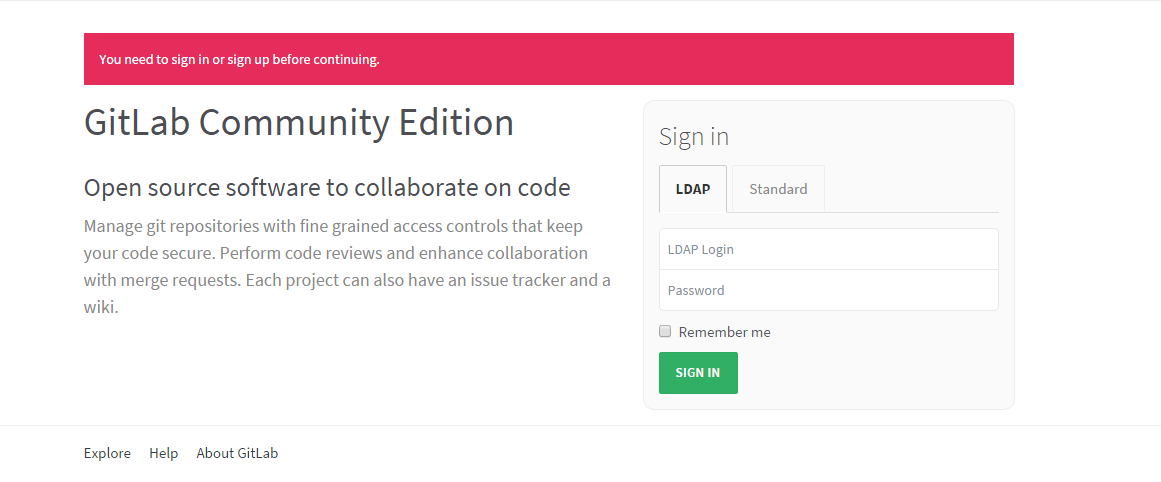
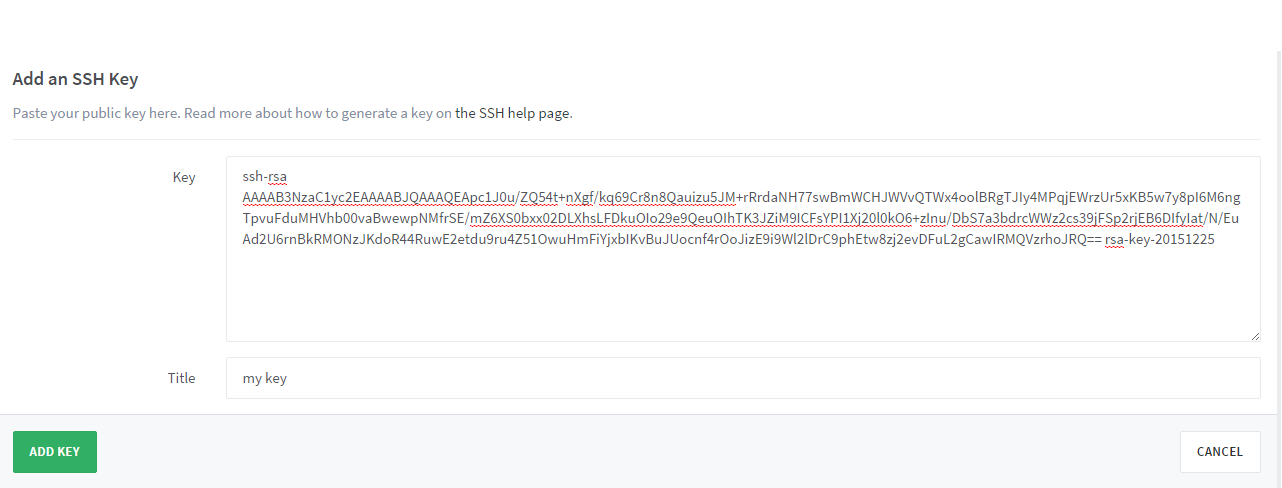
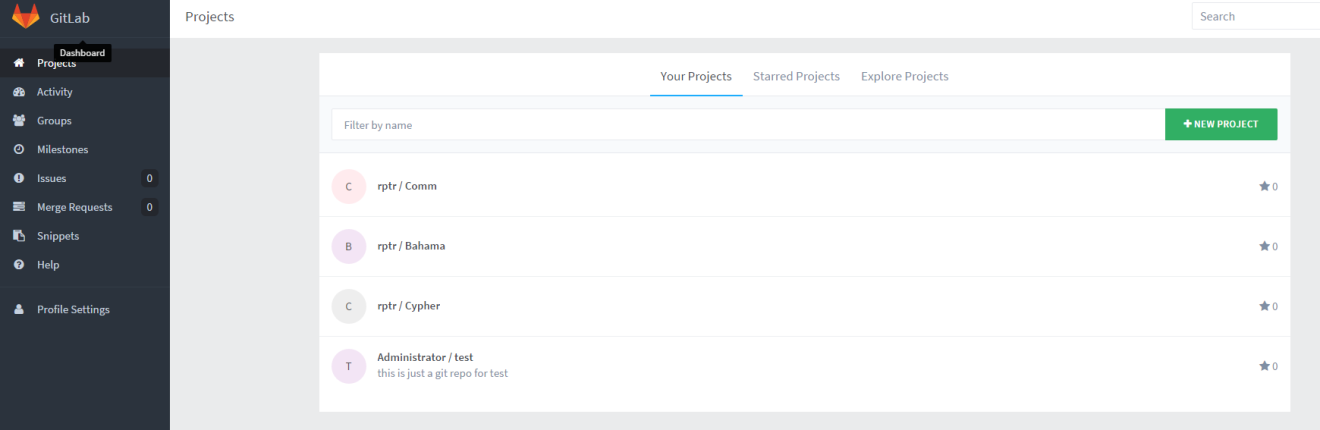
Open the TortoiseGIT setting: Right click any directory -> TortoiseGIT -> Settings

Name: Fu Shanghai-PNW748

Email: shanghai.fu@motorolasolutions.com



## 1.2 Bind your SSH public key

1. Access <http://booster> and found “Gitlab” link in the upper nav bar.
2. Click it, Login with your one IT user/password.
3. Click “Profile Setting”🡪“SSH Keys”🡪”ADD SSH KEY” to paste your public key.
4. Send an email to Bu Hui (GJN374) to apply for access permission. 

## 1.3 Clone GIT repositories

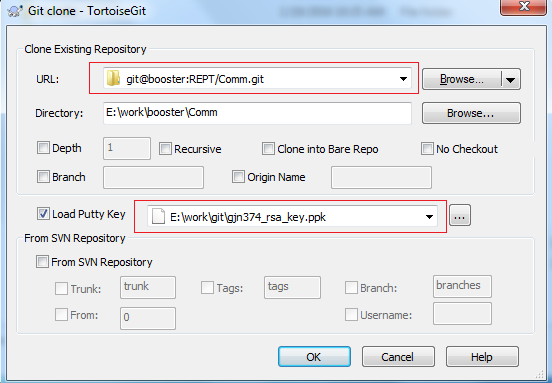
If you already cloned repo before, you can ignore this step and refer section 1.5 to migrate to current repo.

1. Make sure all the following three repositories reside in the same directory of your local PC;
2. Please repeat input below three URLs to clone three repositories in TortoiseGit showed as below diagram. The second red box should select the key file generated in section 1.1.

[git@booster:REPT/Comm.git](mailto:git@booster:REPT/Comm.git)

[git@booster:REPT/Bahama.git](mailto:git@booster:REPT/Bahama.git)

[git@booster:REPT/Cypher.git](mailto:git@booster:REPT/Cypher.git)

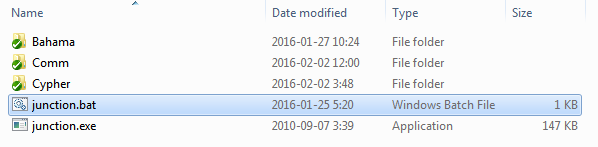


## 1.4 Run junction to link three repo

1. Enter the following shared folder and copy junction.bat and junction.exe to the directory storing your repositories:

\\zch49app08\repeater\Group\50\_Tools\301-RepeaterTools-TCI\Tool Depository\SCM\GIT\Gitlab\

1. Run [junction.bat](file:///\\zch49app08\repeater\Group\50_Tools\301-RepeaterTools-TCI\Tool%20Depository\SCM\GIT\Gitlab\junction.bat) in context of the directory storing all the three repositories.



## 1.5 Migrate from old repo to current repo

If you already clone repo before, open a Git bash console, input below commands to migrate to current repo.

*cmd> cd directory\_of\_Comm // replace the target directory with the real path name*

*cmd> git remote set-url origin* [*git@booster:REPT/Comm.git*](mailto:git@booster:REPT/Comm.git)

*cmd> cd directory\_of\_Bahama // replace the target directory with the real path name*

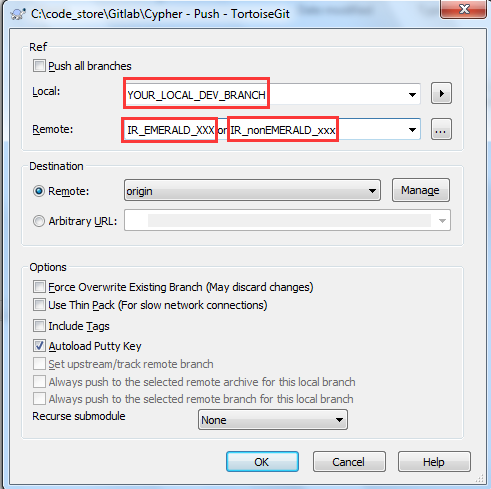
*cmd> git remote set-url origin* [*git@booster:REPT/Bahama.git*](mailto:git@booster:REPT/Bahama.git)

*cmd> cd directory\_of\_Cypher // replace the target directory with the real path name*

*cmd> git remote set-url origin* [*git@booster:REPT/Cypher.git*](mailto:git@booster:REPT/Cypher.git)

## 1.6 How to submit CI

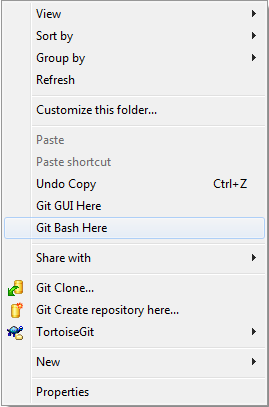
In below first red box, input your local development branch, in second red box, decided by your team, If you from Emerald, remote branch name prefix with **IR\_Emerald**\_XXXX to trigger the CI, If you from NonEmrald, remote branch name prefix with **IR\_NonEmerald**\_XXX to trigger the CI:



# 2. Operate Git with Command in console

## 2.1 Generate SSH Keys and configure

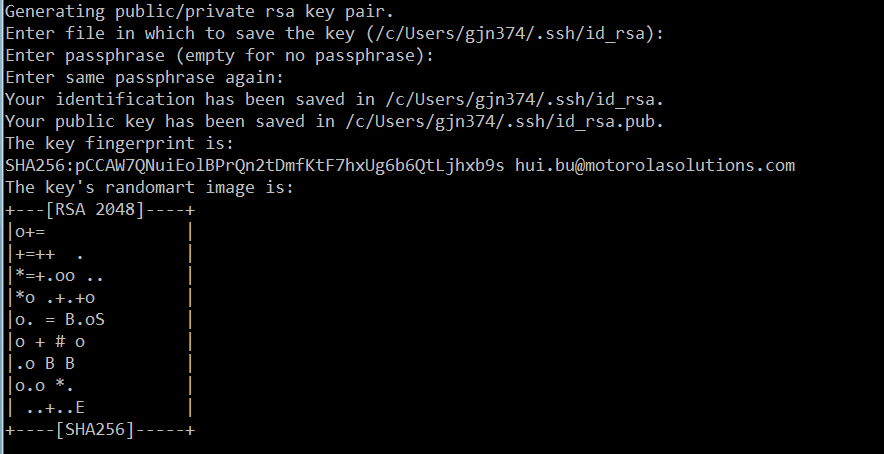
1. Select “Git Bash Here” on the right-click button to open a terminal



1. Run command ssh-keygen -t rsa -C “your company email address”



1. Just press enter to use defaults



1. Done, you’ll see two files in C:\Users\yourcoreid\.ssh\, id\_rsa is private key and id\_rsa.pub is the public key

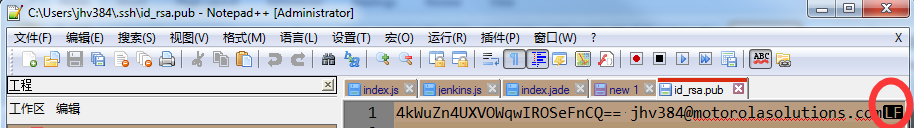


Note:

Different from the public key file generated by TortoiseGit, you should copy all the content of file id\_rsa.pub and bind it to your GIT server account to active your GIT server account.

Caution!!

When you open id\_rsa.pub to copy its content, be care not to include the “LF” character(shown as below red box, you can see it by open the file with notepad++,and enable “see every character” function), otherwise, you will meet issue when you bind your SSH public key in section 2.2.

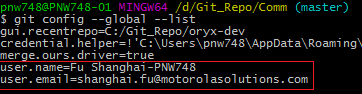


1. Configure username and email

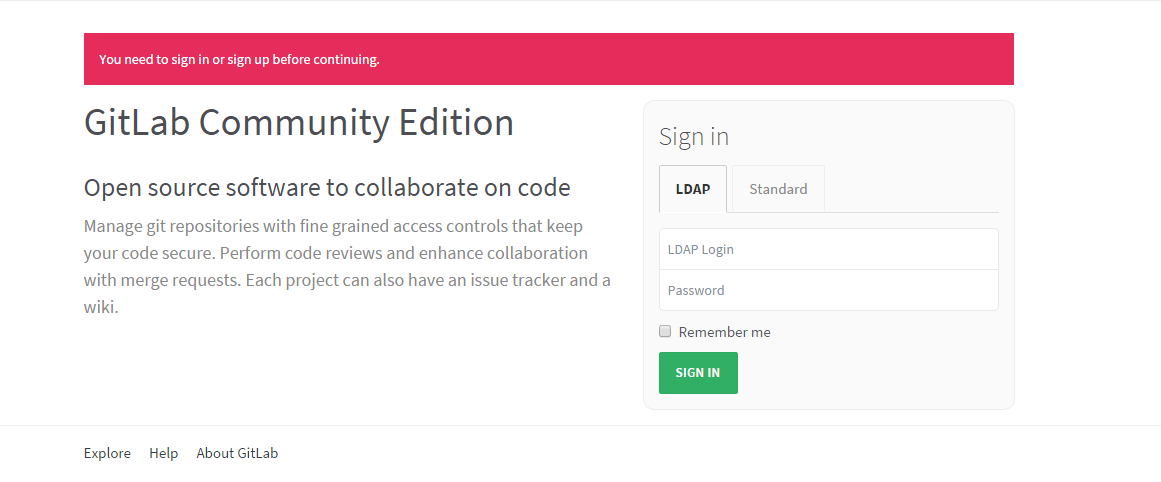
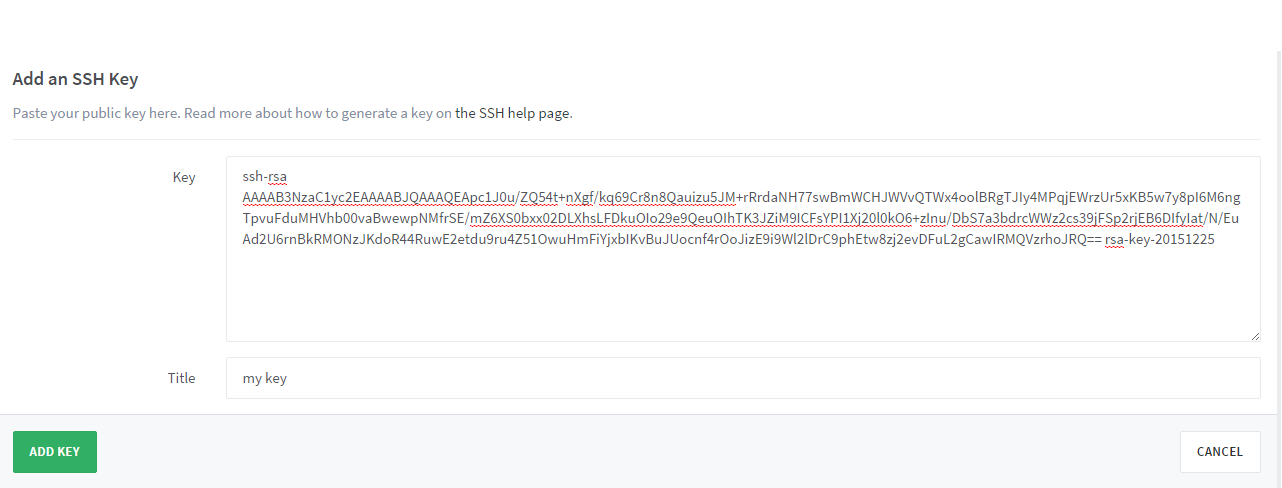
cmd> git config --global user.name "Fu Shanghai-PNW748"

cmd> git config --global user.email [shanghai.fu@motorolasolutions.com](mailto:shanghai.fu@motorolasolutions.com)

You can use ‘git config --global --list’ to check your setting



## 2.2 Bind your SSH public key

1. Access <http://booster> and found “Gitlab” link in the upper nav bar.
2. Click it, Login with your one IT user/password.
3. Click “Profile Setting”🡪“SSH Keys”🡪”ADD SSH KEY” to paste your public key.

Send an email to Bu Hui (GJN374) to apply for access permission.

## 2.3 Clone GIT repositories

1. If you already cloned repo before, you can ignore this step and refer section 2.5 to migrate to current repo.
2. Run below commands in Git Bash console.

*cmd> git clone* [git@booster:REPT/Comm.git](mailto:git@booster:REPT/Comm.git)

*cmd> git clone* [git@booster:REPT/Bahama.git](mailto:git@booster:REPT/Bahama.git)

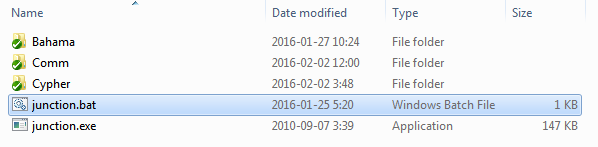
*cmd> git clone* [git@booster:REPT/Cypher.git](mailto:git@booster:REPT/Cypher.git)

## 2.4 Run junction to link three repo

1. Enter the following shared folder and copy junction.bat and junction.exe to the directory storing your repositories:

[\\zch49app08\repeater\Group\50\_Tools\301-RepeaterTools-TCI\Tool Depository\SCM\GIT\Gitlab\](file:///\\zch49app08\repeater\Group\50_Tools\301-RepeaterTools-TCI\Tool%20Depository\SCM\GIT\Gitlab\)

1. Run [junction.bat](file:///\\zch49app08\repeater\Group\50_Tools\301-RepeaterTools-TCI\Tool%20Depository\SCM\GIT\Gitlab\junction.bat) in context of the directory storing all the three repositories.



## 2.5 Migrate from old repo to current repo

If you already clone repo before, open a Git bash console, input below commands to migrate to current repo.

*cmd> cd directory\_of\_Comm // replace the target directory with the real path name*

*cmd> git remote set-url origin* [*git@booster:REPT/Comm.git*](mailto:git@booster:REPT/Comm.git)

*cmd> cd directory\_of\_Bahama // replace the target directory with the real path name*

*cmd> git remote set-url origin* [*git@booster:REPT/Bahama.git*](mailto:git@booster:REPT/Bahama.git)

*cmd> cd directory\_of\_Cypher // replace the target directory with the real path name*

*cmd> git remote set-url origin* [*git@booster:REPT/Cypher.git*](mailto:git@booster:REPT/Cypher.git)

## 2.6 How to submit CI

Open Git Bash console.

* If you from Emerald, remote branch name prefix with IR\_Emerald to trigger the CI:

*cmd> git push origin DEV\_BRANCH\_NAME:IR\_Emerald\_xxx*

* If you from NonEmrald, remote branch name prefix with IR\_NonEmerald to trigger the CI:

*cmd> git push origin DEV\_BRANCH\_NAME:IR\_nonEmerald\_xxx*

## 2.7 Other Gitlab usage: