

Gerrit User Guides

記事本: MiTAC

已建立: 2020/9/29 下午 05:25

已更新: 2020/9/30 下午 04:09

作者: elick.chen@mic.com.tw

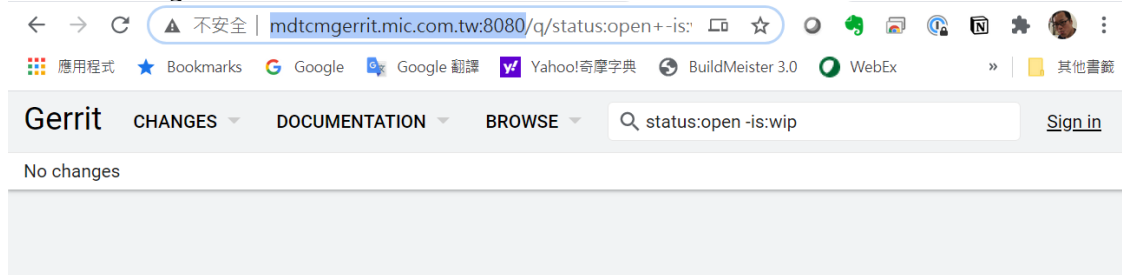
URL: <https://medium.com/@lucienlo/gerrit-%E7%94%B1%E6%B7%BA%E5%85%A5%E6...>

第一次使用Gerrit

MDTCM Gerrit Server: <http://mdtcmgerrit.mic.com.tw:8080/>

MDTCM Gerrit Server 是透過MiTAC AD來進行帳號管理, User 第一次使用, 需要用自己的AD 帳號/密碼 登入上方的URL, 登入成功後會自動建立Gerrit帳號

透過右上角"Sign in" 登入Gerrit



登入成功後, 系統會自動建立Gerrit 帳號, 接下來需要上傳SSH Key

PS: Code review 的協定是走 SSH 29418 port

Generate private / public key

以windows 10 Subsystem for Linux (WSL) 操作為例

```
c:\> wsl
```

```
elick@NMIC48866PC03: /mnt/c/Users/elick.chen
```

```
Microsoft Windows [版本 10.0.19041.450]  
c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。
```

```
C:\Users\elick.chen>wsl  
elick@NMIC48866PC03:/mnt/c/Users/elick.chen$
```

```
# Add new user  
$ sudo adduser elick.chen  
$ sudo su elick.chen  
  
#生成SSH Keys  
$ cd ~/  
$ ssh-keygen -m PEM -t rsa -C "gerrit-user"  
$ cat ./ssh/id_ras.pub
```

將SSH Key加到Gerrit

Setting --> SSH Keys

ELICK.CHEN 陳音佑

elick.chen 陳音佑
elick.chen@mic.com.tw

Settings

Sign out

SSH keys

Comment	Status	Public key
gerrit-admin	Valid	CLICK TO VIEW DELETE

SAVE CHANGES

New SSH key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAC3YaHCgnIQEXQw
Gf2LlibR+niQ8JfhwAR2aFY2RMND1kCJNLIK121yQb4RSGNG
6mh+5IKDmZvDME223W2nQv86vqjeTqD+LmRJqe1QIVu0Go
XcjFKM7QVoA0+OavUFo6P47vKPme0ozUlpGI0+mtyhjHcePL
7V0su1XMo3izKDX3FBgCMXs0YtD0YsglBEj65oM6urSd8bMn
Ux+USRNdZr1DBrK/zw20Ridi4lwK0dEn9yrcYuysrnf77ZS1rXC
2qZvPi0r79deTDr2XvWYHZspRaAXjYS3XK2WwW8icJJZeHK3
OeV6HuOMWvW+xtt/6YukdRuFNZ2mi2I9QfLbdJF gerrit-
admin|
```

[ADD NEW SSH KEY](#)

Groups

Name	Description	Visible to all
------	-------------	----------------

SSH Key 上傳完成後, 通知Gerrit 管理員開通相關權限

安裝git review 工具

使用 Windwos WSL 安裝 git-review 工具

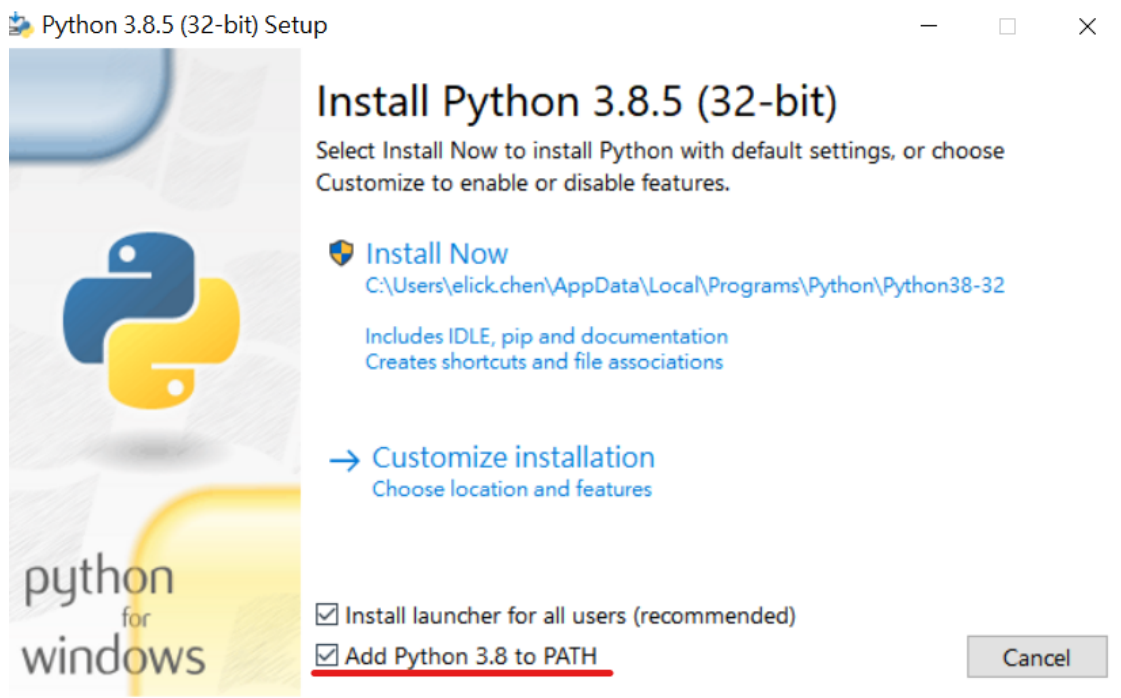
```
Windows 10 WSL(Ubuntu 18.04)
sudo apt update
sudo apt install git-review
```

Windows 10 下安裝 git-review工具

下載安裝 windows python

<https://www.python.org/downloads/>

注意要勾選 "Add Python to PATH"



```
C:\> 命令提示字元

Microsoft Windows [版本 10.0.19041.450]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。

C:\Users\elick.chen>python -V
Python 3.8.5

C:\Users\elick.chen>_
```

利用pip安裝git review工具

```
pip install git-review
```

```
C:\> 命令提示字元

Microsoft Windows [版本 10.0.19041.450]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。

C:\Users\elick.chen>git review --version
git-review version 1.28.0
```

使用 git-review tool, 上傳code到 Gerrit Server, 指令如下

```
git review
```

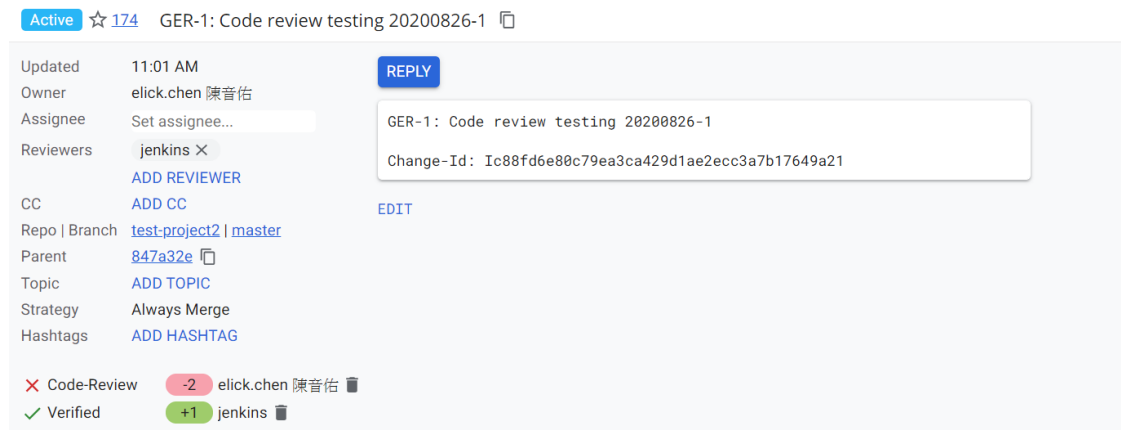
[注意]

如果在多人開發的專案, RD 在code review 前 及 開始coding 前, 最好通過 `git pull --rebase`更新代碼.

Code Review 失敗時處理方式

使用git reset

Code Review不通過時, 可進行暴力回滾: `git reset <commit id>`, 修改 code後再提交Gerrit(git review)



檔 code review 失敗後, 要再提交新的commit到gerrit前, 最快的修正方式時 `git reset`到前一個 commit, 改完code再commit並再提交gerrit(git review)

以下圖為例 "54f35af" commit 的 code review失敗了, 可以 `git reset 847a32e`, 將 "54f35af" 捨棄掉

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
54f35af (HEAD -> master) GER-1: Code review testing 20200826-1
847a32e (origin/master, origin/HEAD, gerrit/master) Merge "GER-1: Jenkins Gerrit Trigger testing-2020082503-1."
190cb4d GER-1: Jenkins Gerrit Trigger testing-2020082503-1.
8f866b0 Merge "GER-1: Jenkins Gerrit Trigger testing-2020082502."
b745ca4 GER-1: Jenkins Gerrit Trigger testing-2020082502.
3a7cfc6 Merge "Jenkins Gerrit Trigger testing-2020082501."
bc4e3da Jenkins Gerrit Trigger testing-2020082501.
bb77122 Merge "Jenkins Gerrit Trigger testing."
37cf1b5 Jenkins Gerrit Trigger testing.
c969d15 Merge "test code review"
6424aca test code review
59794da Merge "test code review"
3527a2e test code review
a0d39ed fix wrong git review config.
4344221 Add git review config.
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$
```

由下圖可看到git reset 後, 上一個commit("54f35af") 被刪除了, 表示可再提交新的commit到gerrit

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git reset 847a32e
Unstaged changes after reset:
M   test-codereview.txt
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
847a32e (HEAD -> master, origin/master, origin/HEAD, gerrit/master) Merge "GER-1: Jenkins Gerrit Trigger testing-2020082503-1."
190cb4d GER-1: Jenkins Gerrit Trigger testing-2020082503-1.
8f866b0 Merge "GER-1: Jenkins Gerrit Trigger testing-2020082502."
b745ca4 GER-1: Jenkins Gerrit Trigger testing-2020082502.
3a7cfc6 Merge "Jenkins Gerrit Trigger testing-2020082501."
bc4e3da Jenkins Gerrit Trigger testing-2020082501.
bb77122 Merge "Jenkins Gerrit Trigger testing."
37cf1b5 Jenkins Gerrit Trigger testing.
c969d15 Merge "test code review"
6424aca test code review
59794da Merge "test code review"
3527a2e test code review
a0d39ed fix wrong git review config.
4344221 Add git review config.
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$
```

採用 git reset方式, gerrit上的 提交的 Change 需要被 abandon

Gerrit CHANGES YOUR DOCUMENTATION BROWSE ACCOUNT PLUGINS

Active ☆ 174 GER-1: Code review testing 20200826-1 REBASE ABANDON EDIT

Updated 11:01 AM
Owner elick.chen 陳音佑
Assignee Set assignee...
Reviewers jenkins X
ADD REVIEWER
CC ADD CC
Repo | Branch test-project2 | master
Parent 847a32e
Topic ADD TOPIC
Strategy Always Merge
Hashtags ADD HASHTAG

REPLY

GER-1: Code review testing 20200826-1
Change-Id: Ic88fd6e80c79ea3ca429d1ae2ecc3a7b17649a21

EDIT

Code-Review -2 elick.chen 陳音佑
Verified +1 jenkins

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ vi test-codereview.txt
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git commit test-codereview.txt -m "GER-1: Code review testing 20200826-2"
[master 2edfaf0] GER-1: Code review testing 20200826-2
1 file changed, 2 insertions(+)
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git review
remote:
remote: Processing changes: refs: 1, new: 1
remote: Processing changes: refs: 1, new: 1, done
remote:
remote: SUCCESS
remote:
remote: http://mdtcmgerrit.mic.com.tw:8080/c/test-project2/+175 GER-1: Code review testing 20200826-2 [NEW]
remote:
To ssh://mdtcmgerrit.mic.com.tw:29418/test-project2.git
* [new branch] HEAD -> refs/for/master
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
2edfaf0 (HEAD -> master) GER-1: Code review testing 20200826-2
847a32e (origin/master, origin/HEAD, gerrit/master) Merge "GER-1: Jenkins Gerrit Trigger testing-2020082503-1."
190cb4d GER-1: Jenkins Gerrit Trigger testing-2020082503-1.
8f866b0 Merge "GER-1: Jenkins Gerrit Trigger testing-2020082502."
b745ca4 GER-1: Jenkins Gerrit Trigger testing-2020082502.
3a7cfc6 Merge "Jenkins Gerrit Trigger testing-2020082501."
bc4e3da Jenkins Gerrit Trigger testing-2020082501.
bb77122 Merge "Jenkins Gerrit Trigger testing."
37cf1b5 Jenkins Gerrit Trigger testing.
c969d15 Merge "test code review"
6424aca test code review
59794da Merge "test code review"
3527a2e test code review
a0d39ed fix wrong git review config.
4344221 Add git review config.
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$
```

使用 git --amend 修補同一個change

當提交的commit未通過時, 再次修改, 可執行下列指令:

```
$ git add .
$ git commit --amend #注意这里会保留上次的 change-id, gerrit 将不会生成新的评审任务编号, 使用原有任务编号, 并将本次提交作为老评审任务的补丁。
$ git review
```

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git commit test-codereview.txt -m "GER-1: Code review testing 20200826-2-fix2"
[master bfe997a] GER-1: Code review testing 20200826-2-fix2
1 file changed, 7 insertions(+), 6 deletions(-)
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git review
remote:
remote: Processing changes: refs: 1, new: 1
remote: Processing changes: refs: 1, new: 1, done
remote:
remote: SUCCESS
remote:
remote: http://mdtcmgerrit.mic.com.tw:8080/c/test-project2/+176 GER-1: Code review testing 20200826-2-fix [NEW]
remote:
To ssh://mdtcmgerrit.mic.com.tw:29418/test-project2.git
* [new branch] HEAD -> refs/for/master
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ vi test-codereview.txt
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git add .
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git commit --amend
[master 0e6d7d4] GER-1: Code review testing 20200826-2-fix2
Date: Wed Aug 26 15:09:09 2020 +0800
1 file changed, 7 insertions(+), 6 deletions(-)
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git review
remote:
remote: Processing changes: refs: 1, updated: 1
remote: Processing changes: refs: 1, updated: 1, done
remote:
remote: SUCCESS
remote:
remote: http://mdtcmgerrit.mic.com.tw:8080/c/test-project2/+176 GER-1: Code review testing 20200826-2-fix2
remote:
To ssh://mdtcmgerrit.mic.com.tw:29418/test-project2.git
* [new branch] HEAD -> refs/for/master
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ |

```

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
0e6d7d4 (HEAD -> master) GER-1: Code review testing 20200826-2-fix2
847a32e (origin/master, origin/HEAD, gerrit/master) Merge "GER-1: Jenkins Gerrit Trigger testing-2020082503-1."
190cb4d GER-1: Jenkins Gerrit Trigger testing-2020082503-1.
8f866b0 Merge "GER-1: Jenkins Gerrit Trigger testing-2020082502."
b745ca4 GER-1: Jenkins Gerrit Trigger testing-2020082502.
3a7cfc6 Merge "Jenkins Gerrit Trigger testing-2020082501."
bc4e3da Jenkins Gerrit Trigger testing-2020082501.
bb77122 Merge "Jenkins Gerrit Trigger testing."
37cf1b5 Jenkins Gerrit Trigger testing.
c969d15 Merge "test code review"
6424aca test code review
59794da Merge "test code review"
3527a2e test code review
a0d39ed fix wrong git review config.
4344221 Add git review config.

```

由下圖可看到patch set number 變為2了

M [test-codereview.txt](#) 

Change Log

Comment Threads



Only comments

elick.chen 陳音佑

Uploaded patch set 1.

jenkins

Added to reviewer: jenkins

jenkins

Build Started <http://mdtbi.mic.com.tw:8080/job/>

jenkins

Verified +1

Build Successful <http://mdtb>

elick.chen 陳音佑

Code-Review -2

elick.chen 陳音佑

Uploaded patch set 2.

jenkins

Build Started <http://mdtbi.mic.com.tw:8080/job/>

jenkins

Verified +1

Build Successful <http://mdtb>

mdtcmgerrit.mic.com.tw:8080/c/test-project2/+176/2/test-codereview.txt

Git工作原則 --

** 永遠是基於遠端庫的最新程式碼工作，儘量每一步操作（特別是add/commit/push）都通過 git pull --rebase獲取一下當前最新版本。 **

```

e1ick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git pull --rebase
Username for 'http://mdtgit.mic.com.tw': elick.chen
Password for 'http://elick.chen@mdtgit.mic.com.tw':
remote: Counting objects: 1, done.
remote: Total 1 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (1/1), done.
From http://mdtgit.mic.com.tw/dev-group/test-project2
   847a32e..3d27d98 master    -> origin/master
Updating 0e6d7d4..3d27d98
Fast-forward
Current branch master is up to date.
e1ick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
3d27d98 (HEAD -> master, origin/master, origin/HEAD) Merge "GER-1: Code review testing 20200826-2-fix2"
0e6d7d4 GER-1: Code review testing 20200826-2-fix2
847a32e (gerrit/master) Merge "GER-1: Jenkins Gerrit Trigger testing-2020082503-1."
190cb4d GER-1: Jenkins Gerrit Trigger testing-2020082503-1.
8f866b0 Merge "GER-1: Jenkins Gerrit Trigger testing-2020082502."
b745ca4 GER-1: Jenkins Gerrit Trigger testing-2020082502.
3a7cfc6 Merge "Jenkins Gerrit Trigger testing-2020082501."
bc4e3da Jenkins Gerrit Trigger testing-2020082501.
bb77122 Merge "Jenkins Gerrit Trigger testing."
37cf1b5 Jenkins Gerrit Trigger testing.
c969d15 Merge "test code review"
6424aca test code review
59794da Merge "test code review"
3527a2e test code review
a0d39ed fix wrong git review config.
4344221 Add git review config.

```

當我們的commit Push到Gerrit等待review時，Gerrit會將此commit儲存在一個名為“refs/changes/xx/yy/zz”的一個暫存branch中。
其中zz為這個commit的patch set號，yy是change號，xx是change號的後兩位。
例如我們工程中的這個大明同學的提交：

<http://review.xxxxx.com:9090/#/c/545/>

一共提交了9次patch，那麼第9個patch就儲存在一個名為“refs/changes/45/545/9”的branch中。

[參考]
<https://codertw.com/ios/56702/>

遇到衝突解決方式

狀況: A user commit code 要提交code到gerrit(git review)前, B user 已提交 gerrit的code review已被submit, 而B user的code 和A user Code 有conflict 所以當A user 下指令要code review時, 會發生衝突


```

e1ick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git review
Errors running git rebase -p -i remotes/gerit/master
Rebasing (1/1)
error: could not apply 20c9b67... test 20200827-090601

Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".
Could not pick 20c9b6797e35b6e6a12488cd4159f8f5ccf455f4
It is likely that your change has a merge conflict. You may resolve it
in the working tree now as described above and then run 'git review'
again, or if you do not want to resolve it yet (note that the change
can not merge until the conflict is resolved) you may run 'git rebase
--abort' then 'git review -R' to upload the change without rebasing.

```

The screenshot displays the GitLab web interface for a project. The top section shows the 'Graph' view of the commit history, with a red line indicating the current branch state. The 'Uncommitted changes (detached head)' section shows the current state of the repository, including the 'develop' branch and the 'HEAD' branch. The 'Staged files' section lists 'new_file1' and 'new_file6' as staged files, and 'test-codereview.txt' as a file with a conflict. The 'Unstaged files' section also lists 'test-codereview.txt' as a file with a conflict.

由上圖可知 "test-codereview.txt"發生了衝突

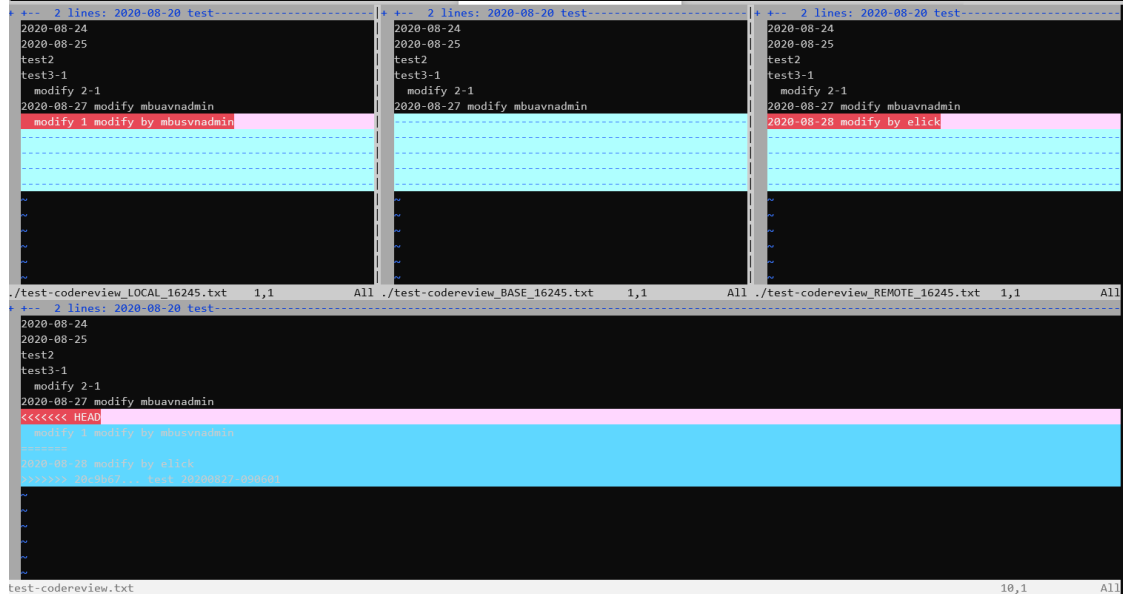
```
test-codereview.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
2020-08-20 test
test2
2020-08-24
2020-08-25
test2
test3-1
    modify 2-1
2020-08-27 modify mbuavnadmin
<<<<<<< HEAD
    modify 1 modify by mbusvnadmin
=====
2020-08-28 modify by elick
>>>>>>> 20c9b67... test 20200827-090601|
```

使用 git mergetool

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git mergetool

This message is displayed because 'merge.tool' is not configured.
See 'git mergetool --tool-help' or 'git help config' for more details.
'git mergetool' will now attempt to use one of the following tools:
tortoisemerge emerge vimdiff
Merging:
test-codereview.txt

Normal merge conflict for 'test-codereview.txt':
  {local}: modified file
  {remote}: modified file
Hit return to start merge resolution tool (vimdiff):
```



```
+/-- 2 lines: 2020-08-20 test-----+
2020-08-24
2020-08-25
test2
test3-1
    modify 2-1
2020-08-27 modify mbuavnadmin
    modify 1 modify by mbusvnadmin
-----+
+/-- 2 lines: 2020-08-20 test-----+
2020-08-24
2020-08-25
test2
test3-1
    modify 2-1
2020-08-27 modify mbuavnadmin
-----+
+/-- 2 lines: 2020-08-20 test-----+
2020-08-24
2020-08-25
test2
test3-1
    modify 2-1
2020-08-27 modify mbuavnadmin
    2020-08-28 modify by elick
-----+
./test-codereview_LOCAL_16245.txt 1,1 All ./test-codereview_BASE_16245.txt 1,1 All ./test-codereview_REMOTE_16245.txt 1,1 All
+/-- 2 lines: 2020-08-20 test-----+
2020-08-24
2020-08-25
test2
test3-1
    modify 2-1
2020-08-27 modify mbuavnadmin
    <<<<<<< HEAD
    modify 1 modify by mbusvnadmin
    =====
    2020-08-28 modify by elick
    >>>>>>> 20c9b67... test 20200827-090601
-----+
test-codereview.txt 10,1 All
```

當 git merge 出現衝突時，直接在 terminal 輸入：

git mergetool

就會開啟 vimdiff 的 mergetool，介面示意如下：

+-----+

```
| LOCAL | BASE | REMOTE |
+-----+
| MERGED |
+-----+
```

- LOCAL: 本機檔案 (也就是現在所在的 branch)
- BASE: local 以及 remote 兩個分支的 base 內容
- REMOTE: 遠端檔案 (要 merge 進來的 branch)
- MERGED: merge 之後的結果, 在這裡解衝突

解衝突就沒什麼特別的, 和平常一樣, 會有許多衝突用 <<<<<< 和 >>>>>> 標示, 解完存檔後記得 commit 就完成 merge 了。比較特別的是使用 vimdiff 當作 mergetool 時, 有一些比較方便的指令可以使用:

" 跳到下一個衝突點[c" 跳到上一個衝突點]c" 取得 buffer 編號:buffers" 從 buffspec / keyword 視窗取得內容:diffget [buffspec|keyword]" 將內容丟至 buffspec / keyword 視窗:diffput [buffspec|keyword]" 刷新 diff 顯示:diffupdate

假設我們有一段程式碼衝突, 希望使用 remote 的內容, 可以使用 diffget 取得 REMOTE 視窗的內容, 取得內容有兩種方式:

1. 使用視窗關鍵字:
:diffget REMOTE

2. 使用視窗編號:

先利用 :buffers 取得視窗編號, 假設想使用的視窗編號為 3:buffers:diffget 3

兩種方式結果都一樣, 但下完指令最好還是確認一下結果是否是自己預期的 XD

如果修改了之後發現 highlight 什麼的有點亂, 可以使用 :diffupdate 來手動刷新一下 diff 顯示

```
+ +-- 2 lines: 2020-08-20 test-----
2020-08-24
2020-08-25
test2
test3-1
  modify 2-1
2020-08-27 modify mbuavnadmin
  modify 1 modify by mbusvnadmin
2020-08-28 modify by elick
~
~
~
~
~
~
~
~
test-codereview.txt [+]
:wq
```

修正衝突後, 存檔離開vimdiff


```

test 20200827-090601

Change-Id: Iacbafe1e8cf16f6a8fe532568edfc5a760b0898b5

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# interactive rebase in progress; onto 2ba0806
# Last command done (1 command done):
#   pick 20c9b67 test 20200827-090601
# No commands remaining.
# You are currently rebasing branch 'develop' on '2ba0806'.
#
# Changes to be committed:
#   new file:   new_file1
#   new file:   new_file6
#   modified:   test-codereview.txt
#
# Untracked files:
#   test-codereview.txt.orig
#

```

編輯 comment 後, 可以看到rebase成功了

```

[detached HEAD 7817f27] test 20200827-090601 - merge mbusvnadmin's code.
 3 files changed, 2 insertions(+)
 create mode 100644 new_file1
 create mode 100644 new_file6
Successfully rebased and updated refs/heads/develop.

```

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
7817f27 (HEAD -> develop) test 20200827-090601 - merge mbusvnadmin's code.
2ba0806 (origin/master, origin/HEAD, gerriit/master) Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
e8a6539 Merge "Add new file #5."
0e5cc5b Merge "Add new file #4."
1f6208f Add new file #5.
21bcda6 Add new file #4.
b353012 Merge "fix issue1"
9a69c39 fix issue1

```

提交code review

\$ git review

vimdiff 常用指令

:diffg RE #get from REMOTE

:diffg BA #get from BASE

:diffg LO #get from LOCAL

:diffupdate #來手動刷新一下 diff 顯示

:wqa #save and exit from vi

使用 vimdiff 在兩份檔案間作比較及複製

按下 Shift + V (進入選擇模式)

使用方向鍵調整要選擇的範圍

按下 y 複製選好的範圍

按下 Ctrl + w 之後按方向鍵移動到要更動的文件

按下 p 貼上複製的內容

按下 esc 以後可輸入指令

do 將另一個檔案的不同內容複製過來

dp 將目前游標所在檔案的不同內容, 複製到另一個檔案

zo 顯示隱藏 / 折疊的文字區塊

zc 將目前游標所在位置文字區塊隱藏 / 折疊

zr 將兩份文件都完全折疊

放棄本地端有衝突的修改, 直接用master上的

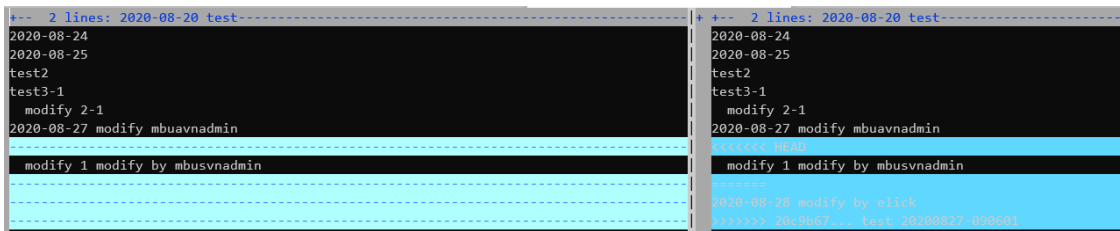
\$ git rebase --skip

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2_1$ git log --oneline
2ba0806 (HEAD, gerriit/master) Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (origin/master, origin/HEAD, master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
e8a6539 Merge "Add new file #5."
0e5cc5b Merge "Add new file #4."
1f6208f Add new file #5.
21bcda6 Add new file #4.
b353012 Merge "fix issue1"
9e69c39 fix issue1
4cd0dd3 Merge "test 20200827_01"
f51c7be test 20200827_01
3d27d98 Merge "GER-1: Code review testing 20200826-2-fix2"
0e6d7d4 GER-1: Code review testing 20200826-2-fix2
0a7732 Merge "GER-1: Code review testing 20200826-2-fix2"
```

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2_1$ git log --oneline
2ba0806 (HEAD -> develop, gerriit/master) Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (origin/master, origin/HEAD, master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
e8a6539 Merge "Add new file #5."
0e5cc5b Merge "Add new file #4."
1f6208f Add new file #5.
21bcda6 Add new file #4.
```

可以看到之前的修改和新增的檔案都沒了

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2_1$ ls
new_file2 new_file3 new_file4 new_file5 new_file_1 test-codereview.txt
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2_1$ ls ../test-project2
new_file1 new_file2 new_file3 new_file4 new_file5 new_file6 new_file_1 test-codereview.txt
```



放棄rebase

這表示不解決衝突了, 等別人來解決完再 rebase

```
$ git rebase --abort
```

```
jellick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2_2$ git rebase --abort
jellick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2_2$ git log --oneline
20c9b67 (HEAD -> develop) test 20200827-090601
3654494 (origin/master, origin/HEAD, master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
e8a6539 Merge "Add new file #5."
0e5cc5b Merge "Add new file #4."
1f6208f Add new file #5.
21bcda6 Add new file #4.
b353012 Merge "fix issue1"
9e69c39 fix issue1
4cd0dd3 Merge "test 20200827_01"
f51c7be test 20200827_01
3d27d98 Merge "GER-1: Code review testing 20200826-2-fix2"
0e6d7d4 GER-1: Code review testing 20200826-2-fix2
847a32e Merge "GER-1: Jenkins Gerrit Trigger testing-2020082503-1."
190cb4d GER-1: Jenkins Gerrit Trigger testing-2020082503-1.
```

狀況: User B commit 前, User A 提交code review 被 submit

```
mbusvnadmin@NMIC48866PC03:~/test/test-project2$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   new_file7
        modified:   test-codereview.txt
```

commit

```
mbusvnadmin@NMIC48866PC03:~/test/test-project2$ git log --oneline
7451d85 (HEAD -> master) test 20200828_144101
2ba0806 (origin/master, origin/HEAD) Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (gerriit/master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
e8a6539 Merge "Add new file #5."
0e5cc5b Merge "Add new file #4."
```

在git review前, User B 下 git pull --rebase, 發現有衝突發生

git pull --rebase

```
mbusvnadmin@NMIC48866PC03:~/test/test-project2$ git pull --rebase
Username for 'http://mdtgit.mic.com.tw': mbusvnadmin
Password for 'http://mbusvnadmin@mdtgit.mic.com.tw':
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 3), reused 0 (delta 0)
Unpacking objects: 100% (5/5), done.
From http://mdtgit.mic.com.tw/dev-group/test-project2
  2ba0806..57c8d0a  master      -> origin/master
First, rewinding head to replay your work on top of it...
Applying: test 20200828_144101
Using index info to reconstruct a base tree...
M       test-codereview.txt
Falling back to patching base and 3-way merge...
Auto-merging test-codereview.txt
CONFLICT (content): Merge conflict in test-codereview.txt
error: Failed to merge in the changes.
Patch failed at 0001 test 20200828_144101
Use 'git am --show-current-patch' to see the failed patch

Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".

mbusvnadmin@NMIC48866PC03:~/test/test-project2$ git log --oneline
57c8d0a (HEAD, origin/master, origin/HEAD) Merge "test 20200827-090601 - merge mbusvnadmin's code."
7817f27 test 20200827-090601 - merge mbusvnadmin's code.
2ba0806 Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (gerriit/master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
```

解決衝突後就可再rebase

```
mbusvnadmin@NMIC48866PC03:~/test/test-project2$ git rebase --continue
Applying: test 20200828_144101
```

```
bcb6adb (HEAD -> master) test 20200828_144101
57c8d0a (origin/master, origin/HEAD) Merge "test 20200827-090601 - merge mbusvnadmin's code."
7817f27 test 20200827-090601 - merge mbusvnadmin's code.
2ba0806 Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (gerriit/master) Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601
```

等一切都沒問題後再進行code review


```
mbusvadmin@NMIC48866PC03:~/test/test-project2$ git review
remote:
remote: Processing changes: refs: 1, new: 1
remote: Processing changes: refs: 1, new: 1, done
remote:
remote: SUCCESS
remote:
remote: http://mdtcmgerrit.mic.com.tw:8080/c/test-project2/+/184 test 20200828_144101 [NEW]
remote:
To ssh://mdtcmgerrit.mic.com.tw:29418/test-project2.git
* [new branch] HEAD -> refs/for/master
```

等User B 的code review成功後, User A 也要 pull 最新 source code下來

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git pull origin master
Username for 'http://mdtgit.mic.com.tw': elick.chen
Password for 'http://elick.chen@mdtgit.mic.com.tw':
From http://mdtgit.mic.com.tw/dev-group/test-project2
* branch master -> FETCH_HEAD
Updating 7817f27..57c8d0a
Fast-forward
Current branch develop is up to date.
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
57c8d0a (HEAD -> develop, origin/master, origin/HEAD) Merge "test 20200827-090601 - merge mbusvadmin's code."
7817f27 test 20200827-090601 - merge mbusvadmin's code.
2ba0806 (gerrit/master) Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 (master) Merge "test 20200827-142601"
```

Gerrit Server上的Merge Conflict

同時有2個code review在進行, 但這2個change都有改同段source code, 只要其中change被submit了, 另一個就會發生衝突

Subject	Status	Owner	Assignee	Repo	Branch	Updated	Size	CR	V
☆ test 20200828_154201	--	elick.chen 陳音佑	--	test-project2	master (develop)	3:42 PM	S		✓
☆ test 20200828_144101	--	mbusvadmin	--	test-project2	master	3:21 PM	XS		✓

Subject	Status	Owner	Assignee	Repo	Branch	Updated	Size	CR	V
☆ test 20200828_144101	Merge Conflict	mbusvadmin	--	test-project2	master	3:21 PM	XS		✓

Merge Conflict ☆ 184 test 20200828_144101

Updated Aug 28
Owner mbusvadmin
Assignee Set assignee...
Reviewers elick.chen 陳音佑 ✕
jenkins ✕
CC ADD REVIEWER
ADD CC

REPLY

test 20200828_144101
Change-Id: I0f00088571530ae23069a6674f428163099ec21a

EDIT

解決方法

Check Gerrit Server 上有問題的branch下來解決

Clone from gerrit

git clone "http://elick.chen@mdtcmgerrit.mic.com.tw:8080/a/test-project2"

Checkout changed branch

\$ git fetch "http://elick.chen@mdtcmgerrit.mic.com.tw:8080/a/test-project2"

refs/changes/84/184/1 && git checkout -b change-184-1 FETCH_HEAD

```

elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git fetch "http://elick.chen@mdtcmgerrit.mic.com.tw:8080/a/test-project2" refs/changes/84/184/1 && git checkout -b change-184-1 FETCH_HEAD
Password for 'http://elick.chen@mdtcmgerrit.mic.com.tw:8080':
remote: Counting objects: 5, done
remote: Finding sources: 100% (3/3)
remote: Total 3 (delta 2), reused 3 (delta 2)
Unpacking objects: 100% (3/3), done.
From http://mdtcmgerrit.mic.com.tw:8080/a/test-project2
 * branch
   refs/changes/84/184/1 -> FETCH_HEAD
Switched to a new branch 'change-184-1'
elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ ls
new_file1 new_file2 new_file3 new_file4 new_file5 new_file6 new_file7 new_file_1 test-codereview.txt
elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git branch
* change-184-1
  master

```

git rebase origin/master

```

elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git rebase origin/master
First, rewinding head to replay your work on top of it...
Applying: test 20200828_144101
Using index info to reconstruct a base tree...
M       test-codereview.txt
Falling back to patching base and 3-way merge...
Auto-merging test-codereview.txt
CONFLICT (content): Merge conflict in test-codereview.txt
error: Failed to merge in the changes.
Patch failed at 0001 test 20200828_144101
Use 'git am --show-current-patch' to see the failed patch

Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".

```

fix 衝突

\$ git mergetool

\$ git add .

\$ git rebase --continue

```

elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git add .
elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git rebase --continue
Applying: test 20200828_144101
elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git log --oneline
a661791 (HEAD -> change-184-1) test 20200828_144101
ea9b9df (origin/master, origin/HEAD, master) Merge "test 20200828_154201"
ab1da22 test 20200828_154201
57c8d0a Merge "test 20200827-090601 - merge mbusvnaadmin's code."
7817f27 test 20200827-090601 - merge mbusvnaadmin's code.
2ba0806 Merge "test 20200827-185301"
13464f8 test 20200827-185301
3654494 Merge "test 20200827-142601"
ae6a3b0 test 20200827-142601

```

\$ git commit --amend -m "<message with Change-Id>"

```

elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git commit --amend -m "test 20200828_14410-fix.
>
> Change-Id: I0f00088571530ae23069a6674f428163099ec21a"
[change-184-1 43a7299] test 20200828_14410-fix.
Author: mbusvnadmin <mbusvnadmin@mic.com.tw>
Date: Fri Aug 28 14:41:48 2020 +0800
3 files changed, 8 insertions(+), 3 deletions(-)
create mode 100644 new_file7
elick@NMIC48866PC03:/mnt/d/test/test/gerrit/test-project2$ git push origin HEAD:refs/for/master
Password for 'http://elick.chen@mdtcmgerrit.mic.com.tw:8080':
Counting objects: 4, done.
Delta compression using up to 6 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 490 bytes | 490.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3)
remote: Processing changes: refs: 1, updated: 1, done
remote:
remote: SUCCESS
remote:
remote: http://mdtcmgerrit.mic.com.tw:8080/c/test-project2/+184 test 20200828_14410-fix.
remote:
To http://mdtcmgerrit.mic.com.tw:8080/a/test-project2
 * [new branch] HEAD -> refs/for/master

```

\$ git push refs/for/master

Active ☆ 184 test 20200828_14410-fix.

Updated 9:43 AM
Owner mbusvnadmin
Uploader elick.chen 陳音佑
Committer elick.chen
Assignee Set assignee...
Reviewers elick.chen 陳音佑 ×
jenkins ×
ADD REVIEWER
CC ADD CC
Repo | Branch test-project2 | master
Parent ea9b9df
Topic ADD TOPIC
Strategy Always Merge
Hashtags ADD HASHTAG

REPLY

test 20200828_14410-fix.
Change-Id: I0f00088571530ae23069a6674f428163099ec21a

EDIT

Code-Review No votes.
Verified No votes.

解決步驟:

- 1) Clone the Gerrit repository
git clone <https://USER@GERRIT-SERVER/a/REPO-FULL-PATHNAME>
- 2) Go to the change page on Gerrit and copy the checkout patch command
git fetch <https://USER@GERRIT-SERVER/a/REPO-FULL-PATHNAME>
refs/changes/XX/YYYYY/Z && git checkout FETCH_HEAD
- 3) Rebase the change
git rebase origin/BRANCH
- 4) Solve the conflicts
git mergetool
- 5) Continue the rebase
git rebase --continue

6) Repeat the steps 4 and 5 until the end of conflicts

7) git commit --amend

Note: Keep the same Change-Id

8) Send the new patchset to Gerrit

git push origin HEAD:refs/for/BRANCH

方法二

The screenshot shows a Gerrit merge conflict page. At the top, it says "Merge Conflict" with a star icon and the number 187, followed by the change ID "test 20200831-152201". Below this, there's a table with details: "Updated" at 3:27 PM, "Owner" mbusvnadmin, "Assignee" Set assignee..., "Reviewers" jenkins X, "CC" ADD CC, "Repo | Branch" test-project2 | master, and "Parent" 5bfa7d6. To the right of the table is a "REPLY" button. Below the table, there's a text input field containing "test 20200831-152201" and a "Change-Id: I026ed1fa8d436302256e4b9c7e4d8c08c3c9e80f" field. There are also "ADD REVIEWER" and "EDIT" links.

Cherry-Pick then submit a new commit

適用於保持 乾淨的 master 分支(所以開發都在local branch下 coding)

User A

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
5bfa7d6 (HEAD -> master, origin/master, origin/HEAD, gerrit/master) Merge "test 20200828_14410-fix."
43a7299 test 20200828_14410-fix.
ea9b9df Merge "test 20200828_154201"
ab1da22 test 20200828_154201
57c8d0a Merge "test 20200827-090601 - merge mbusvnadmin's code."
7817f27 test 20200827-090601 - merge mbusvnadmin's code.
2ba0806 Merge "test 20200827-185301"
```

\$ git checkout master

\$ git pull --rebase

```
elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git log --oneline
82c6af9 (HEAD -> master, origin/master, origin/HEAD) Merge "test 20200831-152601"
cc769b1 (develop) test 20200831-152601
5bfa7d6 (gerrit/master) Merge "test 20200828_14410-fix."
43a7299 test 20200828_14410-fix.
ea9b9df Merge "test 20200828_154201"
ab1da22 test 20200828_154201
57c8d0a Merge "test 20200827-090601 - merge mbusvnadmin's code."
```

cherry-pick for gerrit change 187

\$ git fetch "http://elick.chen@mdtcmgerrit.mic.com.tw:8080/a/test-project2" refs/changes/87/187/1 && git cherry-pick FETCH_HEAD

```
elick@NMIC48866PC0:/mnt/d/test/test/gitlab/test-project2$ git status
On branch master
Your branch is up to date with 'origin/master'.

You are currently cherry-picking commit 5963217.
  (fix conflicts and run "git cherry-pick --continue")
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Changes to be committed:

    deleted:    test-codereview.txt.orig

Unmerged paths:
  (use "git add <file>..." to mark resolution)

    both modified:    test-codereview.txt
```

7 lines: 2020-08-20 test	7 lines: 2020-08-20 test	7 lines: 2020-08-20 test
<pre> 2020-08-27 modify mbusvadmin modify 1 modify by mbusvadmin 2020-08-28 modify by elick modify 2 modify by elick 2020-08-28 modify by mbusvadmin modify 20200828_1432 2020-08-31 modify by elick </pre>	<pre> 2020-08-27 modify mbusvadmin modify 1 modify by mbusvadmin 2020-08-28 modify by elick modify 2 modify by elick 2020-08-28 modify by mbusvadmin modify 20200828_1432 </pre>	<pre> 2020-08-27 modify mbusvadmin modify 1 modify by mbusvadmin 2020-08-28 modify by elick modify 2 modify by elick 2020-08-28 modify by mbusvadmin modify 20200828_1432 2020-08-31 modify by mbusvadmin </pre>
<pre> /test-codereview_LOCAL_2144.txt 1,1 +-- 7 lines: 2020-08-20 test 2020-08-27 modify mbusvadmin modify 1 modify by mbusvadmin 2020-08-28 modify by elick modify 2 modify by elick 2020-08-28 modify by mbusvadmin modify 20200828_1432 <<<<<<< HEAD 2020-08-31 modify by elick ===== 2020-08-31 modify by mbusvadmin xxxxxx 50m7217... test 20200831-152201 </pre>	<pre> /test-codereview_BASE_2144.txt 1,1 </pre>	<pre> /test-codereview_REMOTE_2144.txt 1,1 </pre>

```
$ git commit -m
```

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git status
On branch master
Your branch is up to date with 'origin/master'.

You are currently cherry-picking commit 5963217.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Changes to be committed:
  modified:   test-codereview.txt
  deleted:   test-codereview.txt.orig

```

\$ git review

Subject	Status	Owner	Assignee	Repo	Branch	Updated
☆ merge from change 187	--	elick.chen 陳音佑	--	test-project2	master	4:27 PM
☆ test 20200831-152201	Merge Conflict	mbusvnadmin	--	test-project2	master	3:27 PM

修改同一change id來解決merge conflict

In User A gitlab reop:

\$ git checkout master

\$ git pull --rebase

\$ git fetch "http://elick.chen@mdtcmgerrit.mic.com.tw:8080/a/test-project2"
 refs/changes/87/187/1 && git checkout -b change-187-1 FETCH_HEAD

The screenshot shows the GitLab web interface. At the top, there are icons for Commit, Pull, Push, Fetch, Branch, Merge, Stash, Discard, and Tag. Below these, there's a 'WORKSPACE' section with links to File Status, History, and Search. To the right, there's a 'BRANCHES' section with a list of branches. The 'change-187-1' branch is selected. The main area shows a 'Graph' of commits. The graph starts with 'master' (1 commit) and 'merge from change 187'. It then shows 'origin/master' and 'origin/HEAD' merging 'test 20200831-152601'. This is followed by 'develop' (test 20200831-152601), 'gerrit/master' (Merge "test 20200828_14410-fix."), 'test 20200828_14410-fix.', 'Merge "test 20200828_154201"', and 'test 20200828_154201'.

\$ git rebase origin/master

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git rebase origin/master
First, rewinding head to replay your work on top of it...
Applying: test 20200831-152201
Using index info to reconstruct a base tree...
M       test-codereview.txt
Falling back to patching base and 3-way merge...
Removing test-codereview.txt.orig
Auto-merging test-codereview.txt
CONFLICT (content): Merge conflict in test-codereview.txt
error: Failed to merge in the changes.
Patch failed at 0001 test 20200831-152201
Use 'git am --show-current-patch' to see the failed patch

Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".

```

\$ git mergetool

\$ rm test-codereview.txt.orig

\$ git rebase --continue

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git rebase --continue
Applying: test 20200831-152201

```

\$ git commit --amend

or

\$ git commit --amend -m "message with Change-Id"

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git commit --amend -m "fix conflict.
>
> Change-Id: I026ed1fa8d436302256e4b9c7e4d8c08c3c9e80f"
[change-187-1 22adbae] fix conflict.
Author: mbusvnadmin <mbusvnadmin@mic.com.tw>
Date: Mon Aug 31 15:22:40 2020 +0800
2 files changed, 1 insertion(+), 16 deletions(-)
delete mode 100644 test-codereview.txt.orig

```

\$ git review

```

elick@NMIC48866PC03:/mnt/d/test/test/gitlab/test-project2$ git review
remote:
remote: Processing changes: refs: 1, updated: 1
remote: Processing changes: refs: 1, updated: 1, done
remote:
remote: SUCCESS
remote:
remote: http://mdtcmgerrit.mic.com.tw:8080/c/test-project2/+187 fix conflict.
remote:
To ssh://mdtcmgerrit.mic.com.tw:29418/test-project2.git
* [new branch] HEAD -> refs/for/master%topic=change-187-1

```

Subject	Status	Owner	Assignee	Repo	Branch	Updated	Size	CR	V
☆ fix conflict.	--	mbusvnadmin	--	test-project2	master (change-187-1)	4:47 PM	S		✓

REST API

<https://gerrit-documentation.storage.googleapis.com/Documentation/3.2.3/config-accounts.html#external-ids>

https://gerrit-documentation.storage.googleapis.com/Documentation/3.2.3/access-control.html#capability_modifyAccount

<https://gerrit-documentation.storage.googleapis.com/Documentation/3.2.3/rest-api-accounts.html#delete-account-external-ids>
<https://gerrit-documentation.storage.googleapis.com/Documentation/3.2.3/rest-api-projects.html#set-project-description>
<https://gerrit-documentation.storage.googleapis.com/Documentation/3.2.3/dev-rest-api.html>
<https://gerrit-documentation.storage.googleapis.com/Documentation/3.2.3/rest-api.html>