Conflict handling in Git

Jerry is working on the wchar_support branch. He changes the name of the feature and after checking, he repository his changes.

Make changes in the wchar_support branch

Jerry is working on the wchar_support branch. He changes the name of the feature and after checking, he repository his changes.

```
[jerry @ CentOS src] $ git branch
master
* wchar_support
[jerry @ CentOS src] $ git diff
```

The above command will produce the following result:

```
diff --git a / src / string_operations.cb / src / string_operations.c
index 8fb4b00.01ff4e0 100644
--- a / src / string_operations.c
+++ b / src / string_operations.c
@@ -1,7 +1,7 @@
#include
#include
-size_t w_strlen (const wchar_t * s)
+ size_t my_wstrlen (const wchar_t * s)
{
const wchar_t * p = s;
```

After verifying the code, he republished the changes he has made.

```
[jerry @ CentOS src] $ git status -s
M string_operations.c

[jerry @ CentOS src] $ git add string_operations.c

[jerry @ CentOS src] $ git commit -m 'Changed function name'
[wchar_support 3789fe8] Changed function name
1 files changed, 1 insertions (+), 1 deletions (-)

[jerry @ CentOS src] $ git push origin wchar_support
```

The above command will produce the following result:

```
Counting objects: 7, done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 409 bytes, done.
Total 4 (delta 1), reused 0 (delta 0)
To gituser@git.server.com: project.git
64192f9.3789fe8 wchar_support -> wchar_support
```

Make changes in the master branch

Meanwhile in the master branch, Tom also changed the name of the same function and pushed those changes to the master branch.

```
[tom @ CentOS src] $ git branch
* master
[tom @ CentOS src] $ git diff
```

The above command will produce the result:

```
diff --git a / src / string_operations.cb / src / string_operations.c
index 8fb4b00.52bec84 100644
--- a / src / string_operations.c
+++ b / src / string_operations.c
@@ -1,7 +1,8 @@
#include
#include
-size_t w_strlen (const wchar_t * s)
+ / * wide character strlen fucntion * /
+ size_t my_wc_strlen (const wchar_t * s)
{
const wchar_t * p = s;
```

After diff verification, he repository these changes.

```
[tom @ CentOS src] $ git status -s
M string_operations.c

[tom @ CentOS src] $ git add string_operations.c

[tom @ CentOS src] $ git commit -m 'Changed function name from w_strlen to my_w
[master ad4b530] Changed function name from w_strlen to my_wc_strlen
1 files changed, 2 insertions (+), 1 deletions (-)

[tom @ CentOS src] $ git push origin master
```

The above command will produce the following result:

```
Counting objects: 7, done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 470 bytes, done.
Total 4 (delta 1), reused 0 (delta 0)
To gituser@git.server.com: project.git
64192f9.ad4b530 master -> master
```

On the wchar_support branch, Jerry performs the strchr function for the extended character string. After checking, you should repository and push these changes to the wchar_support branch.

```
[jerry @ CentOS src] $ git branch
master
* wchar_support
[jerry @ CentOS src] $ git diff
```

The above command will produce the following result:

```
diff --git a / src / string_operations.cb / src / string_operations.c
index 01ff4e0.163a779 100644
--- a / src / string_operations.c
+++ b / src / string_operations.c
@@ -1,6 +1,16 @@
#include
#include
+ wchar_t * my_wstrchr (wchar_t * ws, wchar_t wc)
+
{
+
while (* ws)
{
+
if (* ws == wc)
+
return ws;
+
++ ws;
+
}
+ return NULL;
+
}
+
size_t my_wstrlen (const wchar_t * s)
{
const wchar_t * p = s;
```

After verification, he repository of these changes.

```
[jerry @ CentOS src] $ git status -s
M string_operations.c

[jerry @ CentOS src] $ git add string_operations.c

[jerry @ CentOS src] $ git commit -m 'Addded strchr function for wide characte
[wchar_support 9d201a9] Addded strchr function for wide character string
1 files changed, 10 insertions (+), 0 deletions (-)

[jerry @ CentOS src] $ git push origin wchar_support
```

The above command will produce the following result:

```
Counting objects: 7, done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 516 bytes, done.
Total 4 (delta 1), reused 0 (delta 0)
To gituser@git.server.com: project.git
3789fe8.9d201a9 wchar_support -> wchar_support
```

Handling conflicts

Tom wants to see what Jerry is doing on his private branch, so he tries to pull the latest changes from the wchar_support branch, but git aborts this operation with the following error message:

```
[tom @ CentOS src] $ git pull origin wchar_support
```

The above command will produce the following result:

```
remote: Counting objects: 11, done.
63Git Tutorials
remote: Compressing objects: 100% (8/8), done.
remote: Total 8 (delta 2), reused 0 (delta 0)
Unpacking objects: 100% (8/8), done.
From git.server.com:project
* branch
wchar_support -> FETCH_HEAD
Auto-merging src / string_operations.c
CONFLICT (content): Merge conflict in src / string_operations.c
Automatic merge failed;  fix các xa và Then commit the result.
```

Handling conflicts

From the error message, it is easy to recognize that there is a conflict in src / string_operations.c. He runs the git command to see more details.

```
[tom @ CentOS src] $ git diff
```

The above command will produce the following result:

```
diff --cc src / string_operations.c
index 52bec84,163a779.0000000
--- a / src / string_operations.c
+++ b / src / string_operations.c
@@@ -1.8 -1.17 +1,22 @@@
#include
#include
++  HEAD
+ / * wide character strlen fucntion * /
+ size_t my_wc_strlen (const wchar_t * s)
++ =======
+ wchar_t * my_wstrchr (wchar_t * ws, wchar_t wc)
+
{
```

```
+
+
while (* ws)
{
if (* ws == wc)
+
return ws;
+
++ ws;
+
}
+ return NULL;
+
}
+
+ size_t my_wstrlen (const wchar_t * s)
++ >>>>>>> 9d201a9c61bc4713f4095175f8954b642dae8f86
{
const wchar_t * p = s;
```

When both Tom and Jerry changed the name of the same function, git was in a disordered state and it asked the user to solve this problem.

Tom decides to keep the function name from Jerry's proposal, but he keeps the comment added by him. After removing the conflict, git diff will look like this:

```
[tom @ CentOS src] $ git diff
```

The above command will produce the following result:

```
diff --cc src / string_operations.c
diff --cc src / string_operations.c
index 52bec84,163a779.0000000
--- a / src / string_operations.c
+++ b / src / string_operations.c
@@@ -1.8 -1.17 +1,18 @@@
#include
#include
+ wchar_t * my_wstrchr (wchar_t * ws, wchar_t wc)
+
{
+
while (* ws)
{
+
if (* ws == wc)
+
return ws;
+
++ ws;
+
}
+ return NULL;
+
```

```
}
+
+ / * wide character strlen fucntion * /
- size_t my_wc_strlen (const wchar_t * s)
+ size_t my_wstrlen (const wchar_t * s)
{
const wchar_t * p = s;
```

When Tom edited the files, he first had to repository these changes and then he could pull those changes.

```
[tom @ CentOS src] $ git commit -a -m 'Resolved conflict'
[master 6b1ac36] Resolved conflict

[tom @ CentOS src] $ git pull origin wchar_support.
```

Tom has resolved the conflict, now the pull operation will succeed.

According to Tutorialspoint

Previous article: Managing branches in Git

Next lesson: Different platforms in Git