

# Git - Handling Conflicts

## Perform Changes in wchar\_support Branch

Jerry is working on the **wchar\_support** branch. He changes the name of the functions and after testing, he commits his changes.

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

The above command produces the following result –

```
diff --git a/src/string_operations.c b/src/string_operations.c
index 8fb4b00..01ff4e0 100644
--- a/src/string_operations.c
+++ b/src/string_operations.c
@@ -1,7 +1,7 @@
#include <stdio.h>
#include <wchar.h>
-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 commits his 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 '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
```

## Perform Changes in Master Branch

Meanwhile in the master branch, Tom also changes the name of the same function and pushes his changes to the master branch.

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

The above command produces the following result –

```
diff --git a/src/string_operations.c b/src/string_operations.c
index 8fb4b00..52bec84 100644
--- a/src/string_operations.c
+++ b/src/string_operations.c
@@ -1,7 +1,8 @@
#include <stdio.h>
#include <wchar.h>
-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 verifying diff, he commits his 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_wc_strlen'
[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 implements strchr function for wide character string. After testing, he commits and pushes his changes to the **wchar\_support** branch.

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

The above command produces the following result –

```
diff --git a/src/string_operations.c b/src/string_operations.c
index 01ff4e0..163a779 100644
--- a/src/string_operations.c
+++ b/src/string_operations.c
@@ -1,6 +1,16 @@
#include <stdio.h>
#include <wchar.h>
+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 verifying, he commits his 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 'Added strchr function for wide character string'
[wchar_support 9d201a9] Added 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
```

Explore our [latest online courses](#) and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

## Tackle 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 the operation with the following error message.

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

The above command produces 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 conflicts and then commit the result.
```

## Resolve Conflicts

From the error message, it is clear that there is a conflict in `src/string_operations.c`. He runs the `git diff` command to view further details.

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

The above command produces 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 <stdio.h>
#include <wchar.h>
++<<<<<< HEAD
+/* wide character strlen function */
```

```
+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;
```

As both Tom and Jerry changed the name of the same function, Git is in a state of confusion and it asks the user to resolve the conflict manually.

Tom decides to keep the function name suggested by Jerry, but he keeps the comment added by him, as it is. After removing the conflict markers, git diff will look like this.

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

The above command produces 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 <stdio.h>
```

```
#include <wchar.h>
+ 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;
```

As Tom has modified the files, he has to commit these changes first and thereafter, he can pull the 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.