Application of Git CAT&DOG Git study 2013 Day 3

Eon Jeong

March 31, 2013

Sit Study Day 3 Application

Eon Jeong

Contents

applying oatches and orks

Maintaining c repository

Other feature
Thanks

Contents

Application

Eon Jeong

Contents

pplying atches and orks

Maintaining a epository

ther features anks

Maintaining a repository

Other features

Applying patches and forks

Thanks

Section 1

Applying patches and forks

Git Study Day 3

Application

Eon Jeong

Contents

applying patches and forks

repository

Other features

hanks

```
#include <stdio.h>
2
    int main() {
         const char format[11] = "Hello %s!\n";
 4
 5
         char buf[80] = "World";
        printf(format, buf);
6
         return 0:
8
9
1
     #include <stdio.h>
     #include <time.h>
3
    int main() {
4
5
         const char format[11] = "Hello %s!\n";
6
         char buf[80]:
7
         time_t now = time(NULL);
         strftime(buf, 20, "%F %T", localtime(&now));
9
        printf(format, buf);
10
         return 0:
11
12
```

How to compare?

```
Eon Jeong
```

```
Cornerii
```

patches and forks

epository

ther teatures

nanks

```
#include <stdio.h>
         #include <time.h>
    4
         int main() {
4
             const char format[11] = "Hello %s!\n";
5
             char buf[80] = "World";
             char buf[80];
     6
             time_t now = time(NULL);
             strftime(buf, 20, "%F %T", localtime(&now));
             printf(format, buf);
6
   10
             return 0;
9
   11
9
   12
```

oatches and forks Maintainina c

Other feature

anks

Show line-by-line difference (mininum edit) between commits or workspace.

- ► Compare file(s) between commits. (2 or *more*)
- and +, not <and >.
- Display revision ID and adjacent common lines.

We can use format-patch to generate patch file from diff.

Manage remote tracked repositories.

- ▶ remote add remote rename remote remove
- remote set-head remote set-branches remote set-url
- remote prune remote update

Applying oatches and forks

epository

Other features Thanks

Join two or more development histories together.

Use history from their past common revision, if possible.

Applying oatches and orks

epository

other features

nanks

Think about pull without merge. Just update references and objects.

Section 2

Maintaining a repository

Sit Study Day 3
Application

Eon Jeong

Contents

applying oatches and orks

repository

Other features

hanks

branch

Git Study Day 3
Application

Eon Jeong

Contents

Applying oatches and orks

epository

Other features

hanks

List, create, or delete codelines.

Eon Jeong

Contents

applying atches al orks

epository

Other features

nanks

List, create, delete or verify a tag object, GPG-signed.

Applying batches and orks

epository

Other features

nanks

Forward commits to updated stream baseline by differences.

atches an orks

epository

Other features

hanks

Revert some commits without stripping them.

Section 3

Other features

Sit Study Day 3 Application

Eon Jeong

Contents

applying oatches and orks

epository

Other features

hanks

Applying batches and orks

viaintaining a epository

Other features

nanks

Transit HEAD to the specified state.

applying atches an orks

epository

Other feature

hanks

Temporarily store current changes and clean workspace.

atches ar orks

epository

Other features

Handle options, for repository or global(user-wide).

repository

Other features

We can specify untracked files by name specs, e.g. for temporary files.

- Wildcard(*) can be used.
- Without /, pplied recursively.
- Files already staged are not affected.

.gitmodules

Git Study Day 3
Application

Eon Jeong

Contents

Applying batches and orks

epository

Other features

Some workspace is required to contain other repository *module* inside, e.g. for library use. cf. submodule

Applying batches ar orks

Maintaining a epository

Other features

nanks

1. CAT

- 2. DOG
- 3. ???
- 4. PROFIT!