

Application of Git

CAT&DOG Git study 2013

Day 3

Eon Jeong

March 31, 2013

Contents

Git Study Day 3
Application

Eon Jeong

Contents

Applying patches and forks

Maintaining a repository

Other features

Thanks

Contents

Applying
patches and
forks

Maintaining a
repository

Other features

Thanks

Section 1

Applying patches and forks

How to compare?

Contents

Applying
patches and
forks

Maintaining a
repository

Other features

Thanks

```
1  #include <stdio.h>
```

```
2  
3  int main() {
```

```
4      const char format[11] = "Hello %s!\n";
```

```
5      char buf[80] = "World";
```

```
6      printf(format, buf);
```

```
7      return 0;
```

```
8  }
```

```
1  #include <stdio.h>
```

```
2  #include <time.h>
```

```
3  
4  int main() {
```

```
5      const char format[11] = "Hello %s!\n";
```

```
6      char buf[80];
```

```
7      time_t now = time(NULL);
```

```
8      strftime(buf, 20, "%F %T", localtime(&now));
```

```
9      printf(format, buf);
```

```
10     return 0;
```

```
11 }
```

[Contents](#)[Applying
patches and
forks](#)[Maintaining a
repository](#)[Other features](#)[Thanks](#)

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

Show line-by-line difference (minimum 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`

Join two or more development histories together.

- ▶ Use history from their past common revision, if possible.

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

Section 2

Maintaining a repository

List, create, or delete codelines.

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

Forward commits to updated stream baseline by differences.

Revert some commits without stripping them.

Section 3

Other features

reset

Git Study Day 3
Application

Eon Jeong

Contents

Applying
patches and
forks

Maintaining a
repository

Other features

Thanks

Transit HEAD to the specified state.

Temporarily store current changes and clean workspace.

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

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

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

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

Thank you.

Git Study Day 3
Application

Eon Jeong

Contents

Applying
patches and
forks

Maintaining a
repository

Other features

Thanks

1. CAT
2. DOG
3. ???
4. PROFIT!