Effective CMake

CMake is code

↳ Use the same principles for CMakeLists.txt and modules as for the rest of your codebase.

Directories

- → Contain a CMakeLists.txt
- → Entry point for the build system generator
- ↳ Subdirectory may be added with add_subdirectory() and must contain a CMakeLists.txt too.

Scripts

- → <script>.cmake files that can be executed with

```
$ cmake -P <script>.cmake
```

- ↳ Not all commands are supported

Modules

- → are <script>.cmake files located in the CMAKE_MODULE_PATH
- ↳ Modules can be loaded with include() command.

★ Commands

command_name(space separated list of strings)

⇒ Scripting commands change state of command processor
- ↳ set variable
- ↳ change behavior of other commands

⇒ Project commands
- ↳ create build target
- ↳ modify build target

→ Command invocation are not expressions.

★ Variables

```
Set(hello world)
message(STATUS "hello, ${hello}")
```

→ Set with the set() command

→ Expand with ${}.

→ Variables and values are strings

→ Lists are ; separated strings

→ CMake variables are not environment variables

→ Unset variables expands to empty string

★ Comments

```
# a single line comment
```

★ Generator Expressions

→ Generator expressions use the $< > syntax.

→ Not evaluated by command interpretor.

→ Evaluated during build system generation.