

## Семинар #4: CMake: Подключение библиотек.

### Часть 1: Наивное подключение зависимостей

Подключение вручную

Подключение с помощью `add_subdirectory`

Подключение с помощью `git submodules`

### Часть 2: Поиски файлов

`find_file`

`CMAKE_PREFIX_PATH`.

`find_program` и `find_library`

### Часть 3: Поиски пакетов, `find_package`

Как просмотреть добавленные таргеты?

`<PACKAGE>_LIBRARIES`

`<PACKAGE>_TARGETS`

Просмотр `IMPORTED_TARGETS`.

### Module mode

Переменная `CMAKE_MODULE_PATH`.

### Config mode

Алгоритм поиска скрипта в Config mode.

`CMAKE_PREFIX_PATH`.

### Опции команды `find_package`

`EXACT`

`QUIET`

`MODULE`

`REQUIRED`

`COMPONENTS`

### Часть 4: `ExternalProject`

`ExternalProject_Add`. Супербилд.

`GIT_REPOSITORY`, `GIT_TAG`.

### Часть 5: `FetchContent`

`FetchContent_Declare`.

`FetchContent_MakeAvailable`.

`GIT_REPOSITORY`, `GIT_TAG`.

## Часть 6: Toolchain-файлы

CMAKE\_TOOLCHAIN\_FILE

CMAKE\_SYSTEM\_NAME – Целевая ОС (Linux, Windows, Android и т.д.).

CMAKE\_C\_COMPILER – Путь к компилятору C

CMAKE\_CXX\_COMPILER – Путь к компилятору C++

CMAKE\_SYSROOT – Системный корень (аналог `-sysroot` в GCC).

CMAKE\_C\_FLAGS

CMAKE\_CXX\_FLAGS

## Часть 7: Использование пакетных менеджеров

### Часть 8: conan

#### Простой проект

```
conan install
```

```
conan list
```

```
conanfile.txt
```

#### Профили

```
conan profile detect
```