

Обзор самых интересных проблем при портировании КОМПАС-3D под Linux

Александр Тулуп
tulup@ascon.ru

C3Days 2023

CMake

- CMAKE_LINK_DEPENDS_NO_SHARED
- CMAKE_LINK_WHAT_YOU_USE

CMAKE_LINK_DEPENDS_NO_SHARED

- Do not relink all depended targets on .so

CMAKE_LINK_WHAT_YOU_USE

```
1 Linking CXX shared library Build/Exe/.../libkHost.so
2 Warning: Unused direct dependencies:
3   /Source/../../Build/Exe/Debug-x64-Linux/libUI_Preview.so
4   /Source/../../Build/Exe/Debug-x64-Linux/libkDrawTool.so
5   /Source/../../Build/Exe/Debug-x64-Linux/libkSys.so
```

MSVC

- `-fms-extensions`
- `-fvisibility-ms-compatible`

MSVC -fms-extensions

```
1 struct foo
2 {
3     void baz() { std::cout << "Base" << std::endl; }
4 };
5
6 struct bar : public foo
7 {
8     // Clang, -fms-extensions
9     void baz() { return __super::baz(); }
10 };
11
12 int main()
13 {
14     bar{}.baz();
15 }
```

MSVC -fvisibility-ms-compat

```
1 // Foo.so
2 struct IBase
3 {
4     virtual void Foo() = 0;
5 };
6 struct Foo : public IBase
7 {
8     virtual void Foo() override {};
9 };
10 // Bar.so
11 int main()
12 {
13     // -fvisibility-ms-compat, -stdlib=libc++
14     ASSERT(dynamic_cast<Foo*>(GetBase()) != nullptr);
15 }
```

MSVC -fvisibility-ms-compat

- -fvisibility=hidden
- -stdlib=libc++

MSVC -fvisibility-ms-compat

- `__attribute__((visibility("default")))`

MSVC -fvvisibility-ms-compat

- -Wweak-vtables
- warning: A has no out-of-line virtual method definitions; its vtable will be emitted in every translation unit

MSVC -fvisibility-ms-compat

- It sets the default visibility to hidden, like -fvisibility=hidden
- Types, but not their members, are not hidden by default.
- The One Definition Rule is relaxed for types without explicit visibility specifications that are defined in more than one shared object: those declarations are permitted if they are permitted when this option is not used.

Link

- `-Wl, --no-undefined`
- `--exclude-libs, ALL`

Link -Wl,--no-undefined

```
1 void Foo();  
2  
3 // -Wl,--no-undefined  
4 // undefined reference to `Foo() '  
5 void Bar()  
6 {  
7     return Foo();  
8 }
```

Link --exclude-libs,ALL

- excludes symbols in all archive libraries from automatic export

Warnings

- `-Werror`
- `-Wold-style-cast`
- `-Wgnu`
- `-Wunreachable-code`
- `-Wunused-function`
- `-Wdelete-abstract-non-virtual-dtor`
- `-Winfinite-recursion ...`

ASAN

- `-fsanitize=address`

Спасибо!



Ссылки

- -fvisibility-ms-compat
- -fms-extensions
- Run-Time Type Information (RTTI)
- Virtual Tables
- LINK_DEPENDS_NO_SHARED
- LINK_WHAT_YOU_USE