Плагины и их разработка

Для расширения функционала платформы можно использовать плагины, которые представляют из себя динамически подгружаемые библиотеки. Каждый плагин реализует функции Get/Set, доступные непосредственно из lua скрипта и механизм, позволяющий генерировать события. Плагин подключается через главный конфигурационный файл:

<Plugin ID=«<имя_плагина>» Type=«library» Name=«<имя_файла_библиотеки>» />

Далее, после успешной загрузки плагина, доступ к нему можно получить из lua:

Get('<имя_плагина>:<какой-то_адрес>')

или

Set('<имя_плагина>:<какой-то_адрес>=<какое-то_значение>')

Написание плагина (Windows)

Создаем DLL, например через Visual Studio. Из DLL должны быть экспортированы функции: Set(...), Get(...) и PollEvents(...). Функции Get/Set вызываются напрямую через lua скрипт. PollEvents вызывается автоматически, перед рендером каждого кадра. Она должна возвратить строку, вида:

«<какой-то_адрес>=<какое-то_значение>»

если надо сгенерировать событие или строку

«[NULL_STRING]»

если не надо.

Заголовочный файл с прототипами этих функций (lmuLib.h):

```
#pragma once
#include <string>

#ifdef LMULIB_EXPORT
  #define LMULIB_API extern "C" declspec(dllexport)
#else
  #define LMULIB_API extern "C" declspec(dllimport)
#endif

#pragma warning(push)
#pragma warning(disable:4190)

LMULIB_API bool Set(const std::string& _param);
LMULIB_API std::string Get(const std::string& _addr);
LMULIB_API std::string PollEvents();

#pragma warning(pop)
```

Написание плагина (Linux)

Все аналогично.

Заголовочный файл (lmuLib.h):

```
#pragma once
#include <string>

extern "C" {
  bool Set(const std::string& _param);
  std::string Get(const std::string& _addr);
  std::string PollEvents();
}
```

Для того, чтобы скомпилировать динамически загружаемую библиотеку, используем примерно такие команды:

```
g++ -std=c++11 -c *.cpp -fPIC
g++ -shared -o libplugin.so -std=c++11 *.o
```

From:
https://wiki.lambda-mu.com/ - **Lambda-Mu Wiki**

Permanent link:
https://wiki.lambda-mu.com/lm2/ce/s_pl?rev=1596550363

Last update: **2020/11/30 14:12**