

Использование C++ DLL в C#

Стальник Е.С.

Белорусский национальный технический университет

Возможность использовать C\C++ библиотеки в приложениях, написанных на C#, предполагает быстрое создание различного рода приложений с большим функционалом. Так как существует масса открытых и свободно распространяемых библиотек со всеми основными алгоритмами, методами решения конкретных задач, данными, ресурсами. Разработчику на относительно молодом языке C# (2000 год) не надо заново писать уже созданные до него решения, а можно воспользоваться готовыми предложениями (если необходимо – с некоторым уровнем редактирования исходной DLL). Например, при создании библиотеки, экспортируемую функцию в ней можно объявить следующим образом:

```
extern "C" __declspec(dllexport) int Add(int a, int b){return a + b; }
```

Импортировать её в C# проекте необходимо так:

```
[DllImport("dll_test_console.dll", EntryPoint = "Add")]
```

```
private static extern int Sub(int a, int b);
```

После чего её можно использовать: `Console.WriteLine(Add(1,2));`

Кроме того, можно использовать библиотеки низкоуровневого доступа самой операционной системы (такие как `kernel32.dll`) для более гибкого использования ресурсов оборудования.

Для безопасного использования неуправляемого кода в C# есть специальный класс `Marshal`, который обеспечивает корректное взаимодействие двух языков.

Также возможно использование промежуточной библиотеки на `Managed C++`, которая поддерживает безопасность кода и может напрямую использоваться в C#.

В то же время есть и обратная связь, когда библиотека, написанная на C#, может использоваться в C++ реализации. Таким образом, поддерживается почти полная совместимость двух популярных языков, которые, со временем, получают всё более узкую специализацию использования, но продолжают поддерживать интегрируемость друг с другом, дабы не ограничивать возможности программистов, использующих как один, так и другой язык.