

**ИСПОЛЬЗОВАНИЕ УНАСЛЕДОВАННЫХ DLL БИБЛИОТЕК
СРЕДСТВАМИ МЕЖПЛАТФОРМЕННЫХ ВЫЗОВОВ**

При разработке программного обеспечения (ПО) широко применяется повторное использование библиотек, которые содержат успешные, проверенные временем решения. Однако не всегда эти решения реализованы на языке, используемом программистом в целевом проекте. Организация взаимодействия разноязыковых программ может осуществляться на этапе компиляции, компоновки или выполнения.

В данной работе рассматривается подход к взаимодействию программ на этапе выполнения для приложений .NET на языке C# и DLL библиотек на языке C, с использованием механизма межплатформенных вызовов. Выбор языков обусловлен популярностью высокоуровневых языков платформы .NET и большим количеством существующих DLL библиотек, пригодных для повторного использования.

Организация межплатформенных вызовов между выполняемым кодом на языках C# и C возможна благодаря специальному сервису – PlatformInvocation Service (PInvoke), который является частью платформы .NET и представляет собой отдельное пространство имен. Трудность использования PInvoke на практике состоит в управлении памятью и обеспечении преобразования типов данных (маршалинг) для передачи, использования и возврата переменных в разных исполняемых средах. Для маршалинга базовых типов в среде .NET предусмотрены аналоги (прототипы) для преобразования, а определяемые типы можно преобразовывать с помощью специальных атрибутов сервиса PInvoke.

Для облегчения описания прототипов функций, вызываемых из DLL библиотек, разрабатываются специальные утилиты, которые генерируют прототип функции на языке C# на основе прототипа исходной функции из DLL библиотеки. Подавляющее большинство утилит направлены на преобразование функций Win32 API путем использования прямой таблицы сопоставления функций, составленной на основе заранее известного состава функций в библиотеках Windows.

В работе исследуется маршалинг базовых и определяемых типов C и C#, построены схемы передачи непреобразуемых типов и база соответствий. Предлагается программное решение, позволяющее отображать прототипы функций на языке C# для вызова функций из наследуемых DLL библиотек на языке C. В основе решения лежит синтаксический анализ, сопоставление типов, выборка решения из созданной базы соответствий. Такой подход предлагает более гибкое решение, позволяющее генерировать прототип для любой C-функции.

Научный руководитель – Авраменко Е.А., к.т.н., доц.