Здравствуйте, уважаемые члены государственной аттестационной комиссии! Меня зовут Антон Ашмарин, тема моей работы «разработка универсальной платформы для интеграции возможностей расширения и автоматизации в .NET приложения».

Для начала я расскажу об актуальности этой темы, затем перейду к постановке задачи, после этого будет проведен краткий обзор существующих решений, затем более подробно я расскажу про реализацию платформы, и в конце подведу итоги.

При разработке современных программных комплексов зачастую возникают задачи, связанные с поддержкой расширения и автоматизации программного обеспечения. Это необходимо для реализации возможности конфигурирования, настройки, переопределения поведения приложения конечным пользователем, а также для расширения функционала приложения и добавления каких-либо новых возможностей.

Чаще всего реализация перечисленных возможностей достигается с помощью поддержки плагинов или скриптов. Поддержка как плагинов, так и скриптов присуща большинству современного программного обеспечения, например офисных пакетов, графических редакторов, средств автоматизированного проектирования и трехмерного моделирования, что доказывает актуальность этой темы. Стоит отметить, что для интеграции поддержки скриптов и плагинов разработчикам ПО как правило приходится использовать различные платформы и технологии, то есть реализация поддержки плагинов и скриптов отделена друг от друга.

Важным отличием плагина от скрипта является тот факт, что плагин, как правило, реализуется и распространяется различными поставщиками ПО, а скрипты в свою очередь, как правило, пишут конечные пользователи. Плагины, как правило, служат для расширения функционала приложения, а скрипты – для автоматизации действий. При этом для реализации поддержки и плагинов и скриптов должны похожие задачи взаимодействия с основным приложением. Из этого факта родилась идея создания универсального подхода, который бы работал единым образом и с плагинами и со скриптами.

Итак, цель работы – реализация платформы для интеграции в ПО возможностей автоматизации и расширения, отвечающей следующим требованиям:

* Поддержка платформы .NET
* Автоматизация и расширение как единое понятие
* Ориентация на конечного пользователя
* Возможность отладки расширений
* Простота интеграции
* Инструментарий для управления расширениями

В процессе исследования были рассмотрены некоторые существующие решений, позволяющие интегрировать в разрабатываемое приложение возможность автоматизации и расширения. В работе подробно описан каждый продукт и сделаны выводы о его плюсах и минусах. На слайде вы видите некоторые из рассмотренных продуктов. В данном случае особняком стоят два решения: VBA и VSTA. При детальном анализе стало понятно, что VBA было действительно довольно универсальным решением, обладающим большим числом конкурентных преимуществ. В свою очередь VSTA была заявлена компанией Microsoft как замена VBA, и именно эта разработка была первоначально интегрирована на реальном проекте (правда, в процессе интеграции были выявлены многие недостатки, существенно затрудняющие использование VSTA и не позволяющие назвать данное решение универсальным).

Рассмотрим основные недостатки существующих решений:

* Решают частные случаи задачи
* Сложны в интеграции
* Имеют неоправданно большой размер и нагромождение невостребованных возможностей
* Сводятся к практически полностью ручной реализации требуемых функций

Помимо этого каждая разработка обладает своими специфичными особенностями и недостатками, однако перечисленные выше проблемы присущи большинству продуктов.

Проанализировав проблемы, не позволяющие использовать в рамках коммерческого проекта текущие разработки, а также интегрировав в тестовой версии продукта одну из разработок (VSTA – опыт интеграции и возникшие проблемы с подробным анализом причин подробно описаны в работе), было решено реализовать собственную платформу, отвечающую сформулированным требованиям. Полученные результаты необходимо было сравнить с существующими разработками.

Для успешной реализации платформы, изучению и решению подлежал ряд вопросов, перечислю основные из них:

* Интеграция со средой разработки. Это необходимо для предоставления конечному пользователю возможности редактирования кода расширения. Кроме того, требовалось реализация поддержки отладки, а так же изменение поведения среду разработки согласно сценариям ее использования в качестве встроенной среды редактирования расширений
* Управление сборками расширений. Так как платформа ориентирована на конечного пользователя, код расширения может часто меняться.
* Взаимодействие приложения и расширений. Пожалуй, одна из самых важных задач, в рамках решения которой требовалось реализовать простой, но в то же время гибкий и эффективный механизм взаимодействия между сборками.
* Интеграция в расширяемое приложение. Требовалось определить сценарии интеграции в уже реализованное приложение и реализовать инструменты для ее облегчения. Так же, в рамках проработки этого вопроса необходимо было продумать средства управления расширениями, которые впоследствии будут частью основного приложения.

На слайде изображены основные компоненты платформы и способы их взаимодействия. Важной составной частью является интегрированная среда разработки, в качестве которой была выбрана среда с открытым исходным кодом SharpDevelop. Эта среда при помощи технологии SDA может управляться программно, а поддержка плагинов и возможностей переопределения стандартного поведения и изменения внешного вида делают ее отличным выбором. Среда разработки, а также модуль интеграции, отвечающий за взаимодействие с основным приложением, работает в отдельном процессе, взаимодействуя с процессом основного приложения с помощью Windows Communication Foundation. Это решение необходимо для реализации поддержки отладки расширений. Сборки расширений после компиляции попадают в изолированное хранилище, после чего загружаются в адресное пространство приложения. Объекты основного приложения взаимодействует с расширениями с помощью .NET Reflection через общие интерфейсы. (Более подробно детали реализации описаны в пояснительной записке.)

Разработанная платформа отвечает сформулированным требованиям, однако помимо этого предоставляет ряд дополнительных преимуществ:

* Относительная простота интеграции в приложение
* Полноценная гибкая среда разработки
* Интегрированный графический интерфейс управления расширениями

Пришло время обсудить результаты работы. Итак,

* Изучены вопросы, связанные с автоматизацией и расширением приложений
* Изучены существующие подходы в данной области
* Предпринята попытка интеграции одного из существующих решений
* Сделаны выводы о преимуществах и недостатках существующих решений и сформулированы требования к новой разрабатываемой платформе
* Разработана новая платформа
* Платформа интегрирована в существующий проект
* Проведён анализ результатов интеграции и сравнение полученной платформы с существующими решениями

Спасибо за внимание, я с готов ответить на ваши вопросы?