

Данное решение сможет работать как автономно, так и на основе различных существующих систем. К примеру, программное обеспечение, основанное на данном подходе, сможет непрерывно анализировать информацию о всех сотрудниках компании, и в случае превышения какого-либо заданного порогового значения, сможет отправлять оповещение менеджеру по персоналу. В случае с анализом информации о новом кандидате, исходная информация может быть получена от самого кандидата посредством формы для подачи резюме.

Ограничением предложенного подхода можно считать его зависимость от точности и правдивости исходных данных, а также сложность внедрения в текущие рабочие процессы компании.

Таким образом можно сделать вывод, что алгоритмы машинного обучения предоставляют широкие возможности для улучшения и автоматизации не только в научной, но и в прикладной сфере.

Список использованных источников:

1. Andrew Ng. Machine Learning // Stanford University Course, 2012
2. Платформа для проведения соревнования по анализу данных Kaggle [Электронный ресурс]. Дата обновления: 25.03.2017. URL: <https://www.kaggle.com/>

ЗАДАЧИ АВТОМАТИЗИРОВАННОЙ РАЗРАБОТКИ СИСТЕМЫ МОДУЛЕЙ РАСШИРЕНИЯ ПРИЛОЖЕНИЙ MICROSOFT OFFICE

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Коваль Д. А.

Серебряная Л. В. – канд. техн. наук, доцент

Автоматизация расширяемого программирования приложений Microsoft Office является актуальной задачей в сфере разработки программного обеспечения. Приложения Microsoft Office предоставляют ряд средств для своего расширения, однако единственный способ создания полноценного модуля расширения – использование COM-интерфейсов, экспортируемых этими приложениями. Существующие системы автоматизации расширяемого программирования приложений Microsoft Office на основе COM-интерфейсов решают задачи автоматизации разработки модуля расширения. Однако, при необходимости разработки системы таких модулей, существующие средства не в состоянии решить поставленную задачу.

Компьютерная обработка данных в современных офисах стала основным видом обработки информации. Ввиду разнородности самих данных существует набор приложений, специализированных под конкретные типы данных. Однако целесообразно использовать не набор отдельных программ, а интегрированные пакеты офисного обслуживания, так как в них реализуется не просто объединение больших автономных программ в пакеты, а их интеграция в прикладные программные комплексы, означающая их полную унификацию – общий пользовательский интерфейс и единообразные подходы к решению типовых задач. На рынке программного обеспечения в сфере прикладных офисных продуктов доминирует пакет приложений Microsoft Office. В состав пакета входит программное обеспечение для работы с различными типами документов – текстами, электронными таблицами, презентациями.

В последнее время наблюдается тенденция к компьютеризации во всех отраслях человеческой деятельности, вследствие чего необходимость в автоматизации механизмов работы с электронными документами возрастает. Помимо возрастающего спроса на средства управления электронными документами, возникают потребности в расширении возможностей этих средств в соответствии со специализированными нуждами конкретных потребителей. Таким образом, расширение приложений Microsoft Office и автоматизация этого процесса является актуальной задачей в сфере разработки программного обеспечения.

Под расширяемостью понимается возможность конструирования таких иерархий модулей, когда каждый модуль добавляет новую функциональность в систему. Расширяемость подразумевает, что добавление модуля возможно без необходимости вносить какие-либо изменения в существующие модули – не должно быть необходимости даже их перекомпилировать [1]. Расширяемая система должна позволять встраивать сторонним разработчикам новую функциональность:

- без привлечения разработчиков системы;
- без модификации исходного кода системы;
- без перекомпиляции, переустановки и перезагрузки существующих модулей системы;
- без согласования с другими сторонними разработчиками.

Приложения Microsoft Office, соблюдая принципы модульности, дают возможность для своего расширения несколькими способами, например через:

- VBA-макросы;
- XLL-модули;
- COM-интерфейсы.

Язык VBA – это диалект языка программирования Visual Basic, встроенный во множество отдельных программ и прикладных пакетов, в том числе и в пакете Microsoft Office. Это позволяет создавать макросы, которые будут автоматически выполнять требуемую последовательность действий за пользователя. VBA-

макросы взаимодействуют с приложениями MicrosoftOffice через их объектные модели. Объектные модели представляются как COM-интерфейсы, к которым VBA-макрос может обращаться для получения доступа к коду приложения.

Достоинства VBA-макросов:

- построены на основе COM-технологии и позволяют использовать все доступные в операционной системе COM-объекты;

- легкие в освоении.

Недостатки VBA-макросов:

- интерпретируемые;

- невысокая производительность;

- проблемы с обратной совместимостью и переносимостью.

XLL-модули – модули расширения для приложения MicrosoftOfficeExcel. Достоинства XLL-модуля:

- построен на основе COM-технологии и позволяет использовать все доступные в операционной системе COM-объекты;

- компилируемый.

- достаточно высокая производительность;

Недостатки XLL-модуля:

- сложность в освоении;

- поддержка только приложения MicrosoftOfficeExcel.

Приложения MicrosoftOffice предоставляют средства для своего расширения путем экспортирования набора COM-интерфейсов. COM-интерфейсы – единственный способ создания полноценного модуля расширения (имеет смысл в автоматизации) и доступа ко всему интерфейсу прикладного программирования приложений Microsoft Office. COM-модуль расширения должен быть зарегистрирован в реестре ОС Windows для каждого использующего его приложения MicrosoftOffice, поскольку для загрузки своих модулей расширения приложение MicrosoftOffice при загрузке в оперативную память сканирует определенные участки реестра ОС на наличие зарегистрированных COM-модулей расширения. В простейшем случае реализация COM-модуля расширения сводится к поддержке экспортируемого приложениями MicrosoftOfficeCOM-интерфейса IDTExtensibility2, через который хост-приложения MicrosoftOfficeуправляют временем жизни модуля расширения (загрузка, выгрузка, обновление и т.д.).

Разработка COM-модуля расширения к приложениям MicrosoftOffice не тривиальная задача ввиду сложности объектной модели этих приложений, внутреннего механизма COM и разработки на языке программирования C++. Существуют две системы, позволяющие облегчить и автоматизировать процесс разработки – системы VisualStudioToolsforOffice и Add-InExpress. Эти системы автоматизированной разработки скрывают низкоуровневые сложности интеграции в хост-приложение, позволяя разработчикам сконцентрироваться на бизнес-логике модуля расширения [2-3].

Но на практике этого оказывается не достаточно. При наличии системы модулей расширения встает задача их интеграции для решения общих задач. Системы VisualStudioToolsforOfficeиAdd-InExpress покрывают сценарии взаимодействия модуля расширения только с хост-приложением. Помимо этого, возникает потребность совместного использования общей для всех модулей функциональности в виде особых модулей-сервисов для устранения дублирования кода и данных в каждом модуле (дублирование кода и данных имеет очень ограниченное применение и почти всегда не осуществляется на практике). Примером общей функциональности может быть аутентификация, журнализация и т.д.

Современные системы автоматизированной разработки модулей расширения приложений MicrosoftOffice никак не решают следующие фундаментальные проблемы, концентрируясь на разработке отдельного модуля расширения, а не целой системы модулей:

- взаимодействие модулей расширения между собой (причем, как в рамках одного приложения, так и между разными приложениями);

- совместное использование сервисов – особых модулей, экспортирующий общую для всех модулей функциональность разными модулями расширения, работающими одновременно в различных приложениях MicrosoftOffice;

- падение производительности при увеличении количества модулей расширения.

В результате анализавозможностей автоматизированной разработки модулей расширения приложений MicrosoftOffice можно сформировать базовые требования к системе, которая решит обозначенные выше задачи и позволит автоматизировать разработку системы модулей расширения:

1. Автоматизировать процесс разработки управляемых (на языке программирования C#) и неуправляемых (на языке программирования C++) модулей расширения и модулей-сервисов.

2. Предоставить возможность взаимодействия модулей расширения как с хост-приложением, так и с другими модулями-расширениями и модулями-сервисами (интеграция в один комбинированный модуль).

3. Производительность хост-приложения не должна линейно зависеть от количества модулей расширения.

Список использованных источников:

1. Никлаус Вирт о культуре разработки ПО // Открытые системы [Электронный ресурс]. – 1998. – Режим доступа: <http://www.osp.ru/os/1998/01/179366/> / Дата доступа: 15.03.15.
2. VSTO и CAB: Интеграция .NET приложения в MicrosoftWord // Habrahabr [Электронный ресурс]. – 2009. – Режим доступа: <http://habrahabr.ru/post/54880/> / Дата доступа: 16.03.2015.
3. Thangaswamy, V. VSTO 3.0 for Office 2007 Programming / V. Thangaswamy – Packt Publishing, – 2009 – 260 p.