

Дочкин Александр Сергеевич

студент

ФГБОУ ВПО «Кузбасский государственный
технический университет им. Т.Ф. Горбачева»
г. Кемерово, Кемеровская область

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ VISUAL STUDIO ДЛЯ ВЗАИМОДЕЙСТВИЯ С MICROSOFT DYNAMICS AX

Аннотация: в статье описывается создания приложений в Visual Studio при помощи компонента Business Connector и его функционал для обеспечения лучшей работы с ERP-системой Microsoft Dynamics AX на предприятии.

Ключевые слова: Microsoft Dynamics AX, компонент, Visual Studio, разработка приложений, ERP-система.

Управление крупными организациями в наше время трудная задача. Это связано с постоянно-изменяющимися условиями работы, бизнеса, удаленностью частей организации друг от друга и разными видами деятельности отделов. Для автоматизации производств имеется большое количество различных ERP-систем. Одна из этих систем была создана корпорацией Microsoft, и это Microsoft Dynamic AX. Она была разработана в 1998 году и к нынешнему году сильно распространилась в развитых странах.

Microsoft Dynamic AX – это система для автоматизации управления предприятием для среднего и крупного сегментов бизнеса, и предоставляет функции управления производством, дистрибуцией, финансами, персоналом, продажами, проектной деятельность, для контроля и анализа бизнеса и пр. AX представляет собой большую сеть, с множеством баз данных, клиентов, серверов и приложений, взаимодействующих друг с другом. Естественно, для работы с такой сетью нужны знания интерфейса системы и специального языка X++, на котором происходит программирование в Axapta. Но если пользователь не знает всех условностей данной системы, или ему необходимо работать вдали от сети предприятия, но доступ к Axapta ему необходим – что делать в таком случае?

Для решения этой проблемы создаются специальные приложения, которые содержат строго ограниченный функционал Ax, для решения конкретных задач, и которые непосредственно подключены к этой системе, что позволяет иметь доступ ко всем данным, изменениям и возможностям.

Разработкой этих приложений можно заниматься в Visual Studio, благодаря специальному компоненту среды разработки Microsoft Dynamic AX – Business Connector. Business Connector позволяет создавать программные приложения, интегрированные с этой системой. Business Connector представляет собой клиент приложения Ax без пользовательского интерфейса, который содержит в себе основные функции, методы и процедуры для доступа к коду X++, логике и модели безопасности системы. Он содержит специальное ядро, которое интерпретирует и выполняет код и предоставляет среду для взаимодействия с хранилищем всех прикладных объектов в Axapta [2].

В Business Connector имеется ряд основных функций, для обеспечения простейшего взаимодействия с таблицами и объектами Microsoft Dynamic AX [1]. Один из таких методов – это LogonAs. Он позволяет подключиться к системе Ax, войти в неё как пользователь и получить доступ к дальнейшей работе с данными, хранящимися в ней. Получив доступ к системе, можно переходить к работе непосредственно с данными. Метод Create позволяет создавать новые записи в таблицах, новые объекты в хранилище, новые таблицы и прочие различные элементы Ax. Начав создание какого-либо объекта, пользователю необходимо заполнить все поля и свойства данного объекта, прежде чем сохранить объект. Если после сохранения нового объекта понадобилось его изменить, или изменить другой объект, имеется метод Update, позволяющий внести и сохранить изменения в любой из объектов системы.

Кроме создания и изменения, имеются стандартные функции для чтения данных из таблиц, хранящихся в Microsoft Dynamic AX – функция Read, и удаления хранимых данных – функция Delete.

Все эти базовые функции позволяют создавать простые приложения, которые будут выполнять конкретную задачу и в определенных рамках. В них

можно создать простой и понятный интерфейс с помощью элементов интерфейса, имеющихся в Visual Studio.

В чем преимущество создания таких приложений? Во-первых, они создаются для людей не знакомых с принципами работы в Ax, не знающих языка X++, но которые должны вносить различные данные в таблицы и хранилища этой системы. Во-вторых, эти приложения позволяют четко разделить функционал системы по задачам – каждое приложение будет выполнять конкретные задачи, не будет затрагивать другие модули и проекты по незнанию. Кроме этого, вероятность того, что пользователь по незнанию нанесет какой-либо ущерб или вред Ax сводится к нулю, так как все необходимые действия, которые он должен совершить в процессе своей работы, будут представлены в конкретном приложении. И в-третьих, это повышает скорость исправления ошибок в системе, так как каждое приложение связано с конкретными операциями и данными Ax, и найти ошибку будет проще, чем искать их во всей базе.

Список литературы

1. .NET Business Connector Overview [AX 2012] // MSDN – сеть разработчиков Microsoft [Электронный ресурс]. – Режим доступа: <https://msdn.microsoft.com/ru-ru/library/aa659581.aspx> (дата обращения: 01.05.2016).
2. Архитектура Business Connector // Microsoft Dynamics AX URL [Электронный ресурс]. – Режим доступа: <http://dynamicbusinesssolutions.ru/aximpguide.ru/html/492d0d1c-5697-482a-9893-54a06756a4a2.htm> (дата обращения: 02.05.2016).