

УДК 681.3.01:681.324

А.Н. Клименко, Д.В. Борисенко

Национальный технический университет «ХПИ», Харьков

## КОНТРОЛЬ И УПРАВЛЕНИЕ РАЗМЕЩЕНИЕМ ОБЪЕКТОВ

*В статье рассмотрены вопросы реализации и применения новой технологии Adobe AIR в организации учета и отображения автомашин на стоянке. Использование механизма Flash, который встроен в среду AIR, дает возможность создавать удобные, информативные и гибкие интерфейсы для приложений, что было бы затруднительно при написании таких приложений на языке C++. Программный интерфейс реализован в виде однооконной прикладной программы, которая содержит элементы управления для выполнения непосредственных команд управления и размещением.*

**Ключевые слова:** размещение объектов, контроль, Flash, технологии Adobe AIR.

### Введение

Автоматизированная организация контроля и управления размещением объектов в сложных динамических системах с неконтролируемым потоком

данных является актуальной прикладной научно-технической задачей, в которой существенную роль играет организация удобных, информативных и гибких интерфейсов для приложений. Это затрудни-

тельно при написании таких приложений на языке C++, поэтому необходима более приспособленная программная среда, в частности, среда, в которой реализована новая технология Adobe AIR [1 – 4].

**Целью данной статьи** является анализ вопросов реализации и применения новой технологии Adobe AIR на примере организации учета и отображения автомашин на стоянке.

### Результаты исследований

Немного о терминологии, технологии и сути проблемы: напомним вкратце механизмы функционирования „тонких клиентов”, которым является наш AIR клиент. Тонкий клиент – это компьютер-клиент сети с клиент-серверной архитектурой, который переносит все задачи по обработке информации на сервер. Примером тонкого клиента может служить компьютер с браузером, использующийся для работы с веб-приложениями. В нашем случае AIR клиент устанавливается непосредственно на стоянке, данные с этого клиента поступают в локальную сеть или через Интернет к серверу, который может быть размещен где угодно. Технология AIR (Adobe Integrated Runtime) – это кодовое название для независимой от операционной системы (кроссплатформенной) среды запуска, созданной компанией Adobe. Adobe AIR выпущен в конце 2007 г. Разработчики с помощью AIR могут полноценно использовать свои навыки в создании приложений, используя Flash, Flex, HTML, JavaScript, AJAX для разработки и переноса RIA (Rich Internet Applications – насыщенные интернет-приложения) на рабочий стол. Основная черта AIR-приложений – это умение хранить работоспособность за пределами зависимости от наличия соединения с Интернетом. Как и в

случае с локальными приложениями, AIR-приложения способны сохранять информацию на компьютере пользователя и взаимодействовать с ней, пользователь может открывать в них файлы и т.п. Когда пользователь входит в сеть, AIR-приложение действует так же, как онлайн-овое приложение, обновляя информацию и открывая доступ к каким-либо удаленным ресурсам. Когда пользователь отключается, AIR-программа продолжает работать, ожидая следующего подключения, чтоб синхронизировать данные.

Приложение может быть создано при использовании связок:

- Flash / Flex / ActionScript;
- HTML / JavaScript / CSS / Ajax;
- PDF может быть использован в любой из связок.

Результатом может быть приложение:

- основанное на Flash или Flex; основным содержанием его будет Flash/Flex (SWF);
- основанное на Flash или Flex с HTML;
- основное содержимое – HTML, JS, CSS/HTML с Flash/Flex или PDF.

AIR содержит три механизма для преобразования AIR кода (рис. 1). AIR строится на трех компонентах, соединяющих ПК с Интернетом: компонент для рендеринга HTML Webkit, движок Flash и база данных SQLite-Lite. По своим возможностям Webkit сопоставимый с Gecko (Mozilla). На его основе построен браузер Apple Safari. И это удачный выбор: Webkit обеспечивает достаточную производительность и преобразует код в точном соответствии со стандартами HTML.

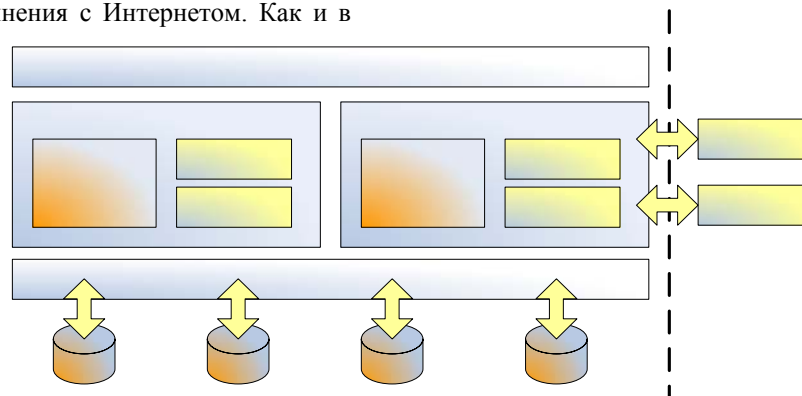


Рис. 1. Схема взаимодействия AIR-приложения

Использование AIR оправдано для:

- систем администрирования сайтов на флеш/флекс – по сравнению с браузерным вариантом получаем работу с файловой системой и драгдроп;
- приложений, тесно интегрированных с веб: мессенджеры и системы общения различных типов, т. к. они рассчитаны на широкую аудиторию, которая любит скины, т.е. флеш- как нельзя кстати;
- приложений, которые должны уметь работать как онлайн, так и оффлайн, с синхронизацией при

появлении интернет-соединения. SQLite – удобный вариант. Это важно для бизнес-приложений;

- так как полноэкранный режим в AIR не имеет функциональных ограничений браузерного, то AIR вполне применим для kiosk-mode тачскрин-интерфейсов (без клавиатуры, если с клавиатурой – то с цинком меньше проблем);
- маленькие приложения-виджеты для повседневного использования;
- простая обработка данных плюс оригиналь-

ный внешний вид и высокая скорость разработки при достаточно обширных функциональных возможностях делают AIR оптимальным выбором [1].

Разработка приложения выполняется на языке Action Script 3. ActionScript 3 – язык программирования для рабочей среды Flash Player'a, который также используется для рабочей среды AIR. Он делает возможным интерактивность, обработку данных и большое количество других функций в среде и приложениях Flash, AIR. ActionScript выполняется интерпретатором ActionScript. Код ActionScript компилируется в байткод с помощью компилятора, например в среде разработки Flex. Байткод переносится в файлы SWF, которые выполняются в рабочей среде Flash Player'a. ActionScript 3.0 предлагает надежную, устойчивую к ошибкам модель программирования, которая привычна для разработчиков, знакомых с объектно-ориентированным программированием. Несколько ключевых деталей ActionScript 3.0 включают в себя следующее: новый интерпретатор ActionScript, названный AVM2, который использует новый набор инструкций в байткоде и предоставляет существенный прирост производительности при его выполнении, расширенный и улучшенный программный интерфейс приложения (API) с низкоуровневым контролем объектов и правильной ООП-моделью. Модель событий основана на объектной модели документов DOM (Level 3 Events Specification) [2].

Использование механизма Flash, который встроен в среду AIR, дает возможность создавать удобные, информативные и гибкие интерфейсы для приложений, что было бы затруднительно при написании таких приложений на языке C++. Программный интерфейс реализован в виде однооконной прикладной программы, которая содержит элементы управления для выполнения непосредственных команд управления и размещением (рис. 2).

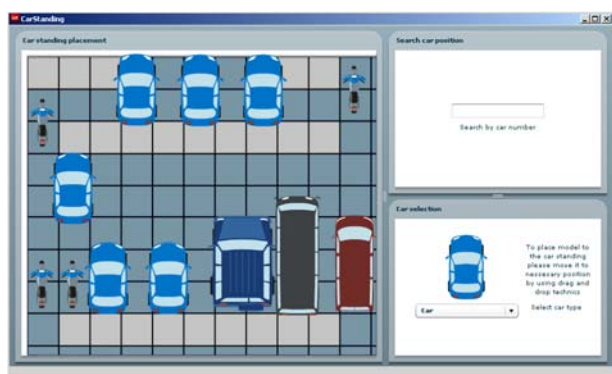


Рис. 2. Окно программного интерфейса

Использована интуитивно понятная технология Drag and Drop. Выходными данными является набор записей базы данных MySQL. Каждая запись отображает состояние стоянки в отдельный момент времени. Таким образом, имеем список автомобильных стоянок, и связанный с ним список измененных состояний для каждой стоянки: Id стоянки, название

стоянки, адрес стоянки, структура стоянки в виде двумерного массива, список машин для каждой стоянки, атрибуты машины (тип машины, координаты размещения, номер машины, время постановки на стоянку, время отъезда, тариф).

При въезде машины на стоянку пользователь вносит информацию о машине (идентификационный номер машины) с помощью приложения AIR, далее приложение автоматически фиксирует время прибытия. Данные передаются веб-серверу, который организован с помощью Apache/PHP. Если нет сообщения с сервером, приложение сохраняет информацию в локальной базе данных SQLite, работая автономно в оффлайновом режиме. При появлении соединения с сервером локальные данные синхронизируются с общей базой данных. Архитектурное решение программного комплекса показано на рис. 3.

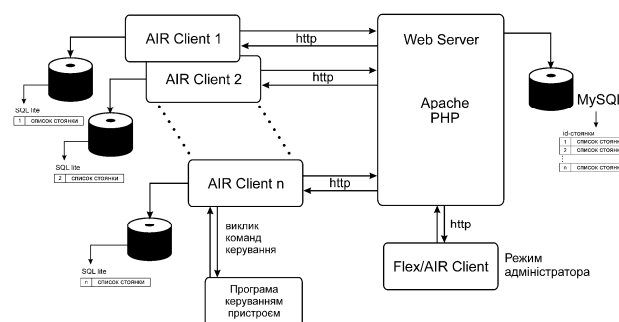


Рис. 3. Архитектурное решение

Передать и принять данные в ActionScript можно разными способами. Один из надежных – это использование объекта XML. Когда мы говорим об XML, всегда подразумевается некоторая валидность документа. В случае AIR это тоже так, но в процессе обмена информацией с сервером, валидация не нужна. Практически любая последовательность символов может быть передана и принята. На основании строки создаем объект XML. Теперь вновь созданный объект XML несет в себе нашу полезную строковую информацию (несмотря на то, что она не является well-formed XML-документом). Её можно вывести на экран методом toString() объекта XML.

Задаем contentType:my\_xml.contentType = "text/xml". Это свойство – гарантия того, что содержимое XML при отправке его на сервер будет размещено в области RAW POST DATA (сырых, не разобранных POST-данных), откуда мы их сможем легко извлечь в php-скрипте. Готовим обработчик получения результата. Как правило, нас мало интересует просто отправка данных, мы хотим быть уверены, что они приняты, обработаны и получен результат. Для этого php должен отправить что-то в ответ, а приложение должно его получить. Получать данные от сервера будет опять-таки объект XML. Вообще, это может быть любой XML ролика, в том числе и тот, который отправлял данные. Ему мы назначим обработчик полученного результата. Для простоты он будет трассировать полученные данные.

Отправляем данные и ждем результат. Для этого воспользуемся методом `SendAndLoad()`. В качестве параметров он принимает название файла, который планируется загружать и объект, в который будут загружены полученные данные. Поскольку задача состоит в передаче данных в `php`, имя этого скрипта и будет в качестве файла, а объект приема данных уже готов. Далее нужно создать скрипт, который смог бы прочесть отправленные данные и выдать ответ. В `php` есть несколько способов получить RAW POST данные. В реальном приложении будет происходить обработка полученных данных, и все прочее, на что способны `php` и программист. Выбор используемых технологий обуславливается следующими параметрами:

1. Гибкость. Возможность расширения функциональности без основательных модификаций приложения, т.е. добавление новых стоянок, изменение схемы стоянки, расширение стоянки.

2. Масштабируемость. Использование большого количества клиентов (стоянок), которые существенно не влияют на производительность.

3. Кроссплатформенность. Возможность развертывания клиента Adobe Air приложения на платформах Microsoft WindowsXP, MacOS, Linux, FreeBSD без дополнительных модификаций.

4. Стоимость. Adobe Air, Apache, PHP, MySQL распространяются бесплатно.

5. Работа клиента в онлайн/офлайн режиме без потери информации.

### Выводы

При разработке насыщенных интернет-программ (RIA) существует несколько вариантов реализации, которые постоянно развиваются и изменяются. Были рассмотрены варианты реализации RIA, среди которых остановились на Flex-Air. Так как Flex имеет средства для разработки Adobe AIR, встроенные в SDK и Builder, то есть возможность создавать Flex приложения размером менее 50К за счет новой возможности Adobe Air Player – кэша для компонентов Adobe platform. Также возможна поддержка рефакторинга, имеются новый профайлер для настройки производительности и использования памяти, а также генератор кода для доступа к источникам данных.

Технология Adobe AIR еще новая, но очень перспективная. Эта технология уже в настоящий момент имеет преимущества в разработке программ, что позволит программам, написанным для ПО отображение информации Adobe Flash, работать без веб-сервера-браузера. Задание AIR – преодолеть ограничение существующих веб-приложений. В настоящий момент Flash-программы работают внутри браузера. AIR – это клиентское ПО, которое позволит выполнять их отдельно от браузера, как в онлайн-режиме, так и в автономном режиме. А также позволит разработчикам создавать дополнения, способные работать как с Flash-анимацией, так и с HTML и файлами Acrobat (PDF). Этот подход хранит преимущества Web, но оставляет место для программ, которые в настоящий момент недоступны. Веб-сервера-приложения, такие как электронная почта на базе Web, могут работать на разных операционных системах, но обычно не работают без соединения с Интернетом. Перспективы для разработчиков такие, что они могут использовать свои навыки в разработке Интернет-приложений для десктопа, например, используя AIR, перенести существующие Интернет-программы для веб-браузеров.

Таким образом, реализация AIR приложения позволяет нам выполнять администрирование сайтов на Flex/Flash, – по сравнению с браузерным вариантом получаем работу с файловой системой и драг-дроп, работа программы как онлайн, так и офлайн, с синхронизацией при появлении интернет-соединения, простая обработка данных и оригинальный внешний вид.

### Список литературы

1. Mike Chambers, Robert L. Dixon, and Jeff Swartz. *Apollo for Adobe Flex Developers*, 2007. – 450 p.
2. Колін Мук. *ActionScript 3.0*, 2007. – 386 p.
3. Электронный ресурс: Режим доступа: <http://www.labs.adobe.com/technologies/air/>
4. Электронный ресурс: Режим доступа: <http://www.labs.adobe.com/technologies/air/>

Поступила в редколлегию 1.04.2008

**Рецензент:** д-р техн. наук, проф. В.А. Краснобаев, Харьковский национальный технический университет сельского хозяйства им. П. Василенко, Харьков.

### КОНТРОЛЬ І УПРАВЛІННЯ РОЗМІЩЕННЯМ ОБ'ЄКТІВ

А.М. Клименко, Д.В. Борисенко

У статті розглянуті питання реалізації і застосування нової технології Adobe AIR в організації обліку і відображення автомашин на стоянці. Використання механізму Flash, який вбудований в середу AIR, дає можливість створювати зручні, інформативні і гнучкі інтерфейси для додатків, що було б важко при написанні таких додатків на мові C++. Програмний інтерфейс реалізований у вигляді одновіконної прикладної програми, яка містить елементи управління для виконання безпосередніх команд управління і розміщенням.

**Ключові слова:** розміщення об'єктів, контроль, Flash, технології Adobe AIR.

### CONTROL AND MANAGEMENT PLACING OF OBJECTS

A.N. Klimenko, D.V. Borisenko

In the article the questions of realization and application of new technology of Adobe AIR are considered in organization of account and reflection of cars on a stand. The use of mechanism of Flash, which is built-in on Wednesday AIR, enables to create comfortable, informing and flexible interfaces for appendixes, that would be difficult at writing of such appendixes in language of C++. A programmatic interface is realized as a one-window application program which contains custom for implementation of direct commands of management controls and placing.

**Keywords:** placing of objects, control, Flash, technologies of Adobe AIR.