

УДК 621.396.218

Ю.Э. Парфенов, В.Н. Федорченко, А.В. Щербаков

Харьковский национальный экономический университет, Харьков

## АНАЛИЗ ОСНОВНЫХ ОСОБЕННОСТЕЙ SILVERLIGHT 4

В статье рассмотрены основные принципы построения технологической платформы от Microsoft – Silverlight 4. Проанализированы существующие среды для разработки кроссплатформенных приложений. Рассмотрена история развития платформы Silverlight. Изложена структура и возможности Silverlight 4 по разработке интерактивных интернет-приложений нового поколения.

**Ключевые слова:** Silverlight, RIA, WPF, XAML, Web 2.0, платформа, пользовательский интерфейс, WCF, инструментальные средства разработки ПО.

### Введение

Silverlight – основанная на Microsoft .NET Framework технологическая платформа, которая обеспечивает IT-профессионалам быструю и рентабельную разработку интерактивных интернет-приложений нового поколения с «богатым» графическим интерфейсом (Rich Internet Applications – RIA). Кроссплатформенные бизнес-приложения на основе Silverlight позволяют максимально удовлетворить требования потребителей и помочь организациям получить максимальный коэффициент возврата инвестиций [1, 2].

В основе функциональной модели Silverlight лежит «клиент-серверная» модель, использующая протокол SOAP во взаимосвязи с технологией WCF (рис. 1).

Silverlight обеспечивает платформу для разработки независимых от конкретного браузера, кроссплатформенных RIA. Все версии Silverlight являются подмножеством WPF – мощной презентационной платформы для создания интерактивных пользовательских интерфейсов, в которые можно легко добавлять медиа-контент (аудио, видео, изображения) и данные. Ядро платформы Silverlight – основанный на XML декларативный язык Extensible Application Markup Language (XAML) [2]. XAML дает возможность дизайнерам и разработчикам независимо друг от друга разрабатывать пользовательские интерфейсы и связанные таблицы стилей. Таким образом, Silverlight – естественное расширение существующих технологий, в частности .NET и WPF. Другими словами, если выбросить ненужные элементы .NET, которые к тому же не являются кроссплатформенными, добавить XAML и несколько новых элементов, таких как функциональная совместимость браузера и способность поддержки динамических языков, например Python, то получим Silverlight – платформу нового поколения для Web-разработки.

**Анализ вопросов разработки Web-приложений.** Разработка приложений, которые работают

на различных платформах, является трудной задачей. Платформа – это уникальная среда, обеспечивающая выполнение некоторого программного кода.

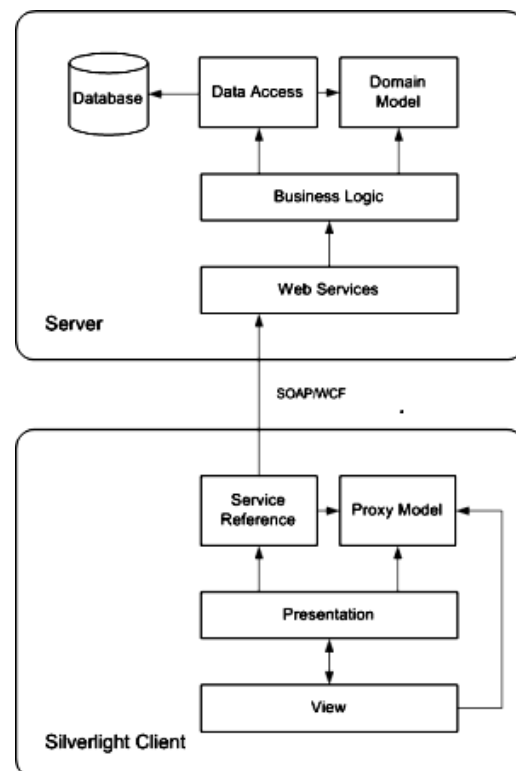


Рис. 1. Функциональная модель Silverlight

Платформами можно считать операционные системы, например Windows 7, Windows Vista, Windows XP, Linux. Web-браузеры, такие как Firefox, Internet Explorer, Opera и устройства, такие как компьютеры и мобильные телефоны серии Windows 7 также представляют собой платформы. Если вы выполняли любую Web-разработку, предназначенную для использования с разными браузерами, то вы знакомы с тем, что обеспечение одинаковой работы Web-сайта в Internet Explorer, Firefox и других браузерах является очень трудной задачей. Цель Silverlight состоит в том, чтобы создать со-

вместимую среду выполнения, независимую от различных браузеров, операционных систем и устройств [3].

Нет никакой причины, чтобы считать, что кроссплатформенное приложение – это автоматически "хорошо". Любая ответственная разработка программного обеспечения начинается с тщательного изучения бизнес-оснований для разработки проекта. Если все пользователи используют единственную платформу, например Windows, не нужно тратить дополнительное время разработки на то, чтобы программное обеспечение также работало и на других платформах.

Однако, кроссплатформенные приложения определено важны в сегодняшнюю эру Web 2.0 создания Web-сайтов, которые пригодны для использования с помощью любого браузера. Способность разрабатывать кроссплатформенные приложения имеет большое значение, когда потенциальные пользователи приложения используют различные платформы. Это довольно очевидное утверждение, но важно отметить, что разработка кроссплатформенного приложения не предлагает никаких преимуществ, если все пользователи находятся на единственной платформе, за исключением случаев, когда межплатформенный аспект является «бесплатным или почти бесплатным» (таким образом, способствуя будущему существованию приложения при изменении круга пользователей). Концепция «бесплатный или почти бесплатный» является важной, так как разработка программного обеспечения сама по себе является непростым предприятием [2,3]. Если придание программному обеспечению кроссплатформенных возможностей является трудноосуществимым, то это потребует или значительно больше времени на разработку. Несомненно, это также означает большие затраты финансов и других ресурсов.

В наилучшем случае, нужен относительно простой способ создания кроссплатформенных приложений. К счастью, есть достаточно много платформ (включая Microsoft Silverlight), которые пытаются сделать создание кроссплатформенных приложений «бесплатным или почти бесплатным».

### Кроссплатформенные программные среды

Программные среды для разработки кроссплатформенных приложений не новы. Даже язык C возможно является кроссплатформенным, так как исходный код может быть написан один раз и откомпилирован на каждой целевой платформе, таким образом обеспечивая мобильность проектов, написанных на C. Хотя споры о том, что действительно представляет собой кроссплатформенность могут быть интересны, они не имеют большого практиче-

ского значения для нас здесь. Поэтому давайте посмотрим на серьезные претенденты для разработки кроссплатформенных приложений.

**Qt.** Qt – инструментарий для разработки кроссплатформенных приложений главным образом для C++; однако, он поддерживает другие языки, такие как Java. Существенная выгода от Qt – то, что программы выполняются в «родной» среде после компиляции (то есть, никакая новая виртуальная машина не нужна). Кроссплатформенная природа Qt обеспечивается на уровне исходного кода, так как разработчики используют независимые от платформы библиотеки Qt. Главные недостатки Qt – трудность изучения для разработчиков и степень выигрыша от его применения (хотя это может быть приемлемым для многих организаций).

**Платформа Java.** Платформа Java (главным образом Java-апплеты и JavaFX) является возможно самым близким аналогом Silverlight на рынке. Во многом подобная .NET, платформа, основанная на Java, является управляемой средой. Однако, до появления Silverlight платформа .NET главным образом была доступна на операционной системе Windows. Обе платформы обеспечивают возможность компиляции программы и ее немедленного выполнения на различных платформах. Подходы платформы Java и Silverlight подобны: среда выполнения (известная как виртуальная машина) разработана для каждой платформы, на которой могли бы выполняться программы. Исходный текст Java компилируется в байт-код java, который затем выполняется виртуальной Java-машиной в среде «песочницы». Недостатки этого подхода – избыток виртуальных машин, которые могут быть созданы, каждая с потенциальными индивидуальными особенностями, которые иногда затрагивают существующие приложения, а также «стоимость» запуска виртуальной Java-машины на Web-сайте. У компании Sun также есть более прямой конкурент Silverlight по имени JavaFX. Это платформа, которая включает язык сценариев, чтобы обеспечить более простое создание Java-приложений. Эта платформа имеет наибольший смысл для учреждений и разработчиков, которые уже привыкли к работе в среде Java или должны расширить их существующие Java-приложения.

**Adobe Flash/Flex/AIR.** Adobe Flash безусловно является самым популярным сравнением с Silverlight. Плагин для браузера, который обеспечивает выполнение «богатого» информационного наполнения для Web – разве это не звучит знакомо? Это сравнение становится еще более точным с выходом Adobe Flex. Это среда для выполнения RIA-приложений в браузере и на настольных компьютерах. Adobe Flex предоставляет богатую библиотеку компонентов с графическим интерфейсом и исполь-

зует MXML – основанный на XML декларативный язык, чтобы разрабатывать интерактивные пользовательские интерфейсы. Хотя есть некоторые различия между Adobe Flex и Silverlight, которые могут повлиять на выбор платформы, Adobe Flex является жизнеспособной альтернативой Silverlight. Однако, Adobe Flex удовлетворяет потребности другого подмножества разработчиков, чем Silverlight. Adobe Flex использует языки, которые разработчики уже знают, включая JavaScript, HTML, CSS, и ActionScript. Silverlight, однако, обеспечивает язык разметки и является невероятно естественной платформой разработки, если вы уже .NET-разработчик.

В дополнение к Adobe Flash и Adobe Flex в феврале 2008, Adobe выпустила Adobe AIR для разработки настольных приложений, которые Вы можете расширить как RIA.

**Microsoft ASP.NET AJAX.** Microsoft ASP.NET AJAX – это набор библиотек JavaScript, встроенный в ASP.NET 3.5 и 4.0. Также доступен как отдельная загрузка для ASP.NET 2.0. Будучи неотъемлемой частью ASP.NET 3.5, 4.0 и библиотеки AJAX, теперь клиентские и серверные библиотеки ASP.NET AJAX более интегрированы с Visual Studio 2010 (и Visual Studio 2008). Клиентская библиотека позволяет Вам реализовать обработку клиентского уровня, например, обработку и проверку допустимости информации, введенной конечным пользователем, регенерацию части web-страницы, и разработку «богатых» интерактивных пользовательских интерфейсов. Вы можете также эффективно интегрировать компоненты клиентской библиотеки с библиотекой серверных элементов управления ASP.NET в асинхронном режиме. Ключевой «двигатель» технологии ASP.NET AJAX – это сценарии. Вообще, основанные на скриптах Web-приложения сталкиваются с несколькими трудностями из-за различных параметров настройки браузера (например, JavaScript может быть отключен по умолчанию). В результате использование сценариев не всегда является лучшей стратегией для предприятий, разрабатывающих безопасные и масштабируемые RIA. ASP.NET AJAX также поддерживает ограниченные черты RIA и не поддерживает эффективную интеграцию мультимедиа, соответствующий управляемый код, и метаданные. Microsoft ASP.NET AJAX – широко распространенная модель для создания RIA, но очень вероятно, что, имея Silverlight как вариант, .NET-разработчики переведут приложения ASP.NET AJAX на платформу Silverlight.

**Microsoft Silverlight.** .NET Framework 3.0 включал первый выпуск WPF, наряду с другими ключевыми технологиями. С WPF прибыл XAML, по существу как способ создания разметки приложений (существует почти взаимно-однозначное соответствие между конструкциями XAML и кодом).

Microsoft Silverlight - подмножество WPF, который является частью .NET Framework 3.x и 4.0. Silverlight интегрирован с широким диапазоном инструментальных средств Microsoft и служб, таких как Microsoft Visual Studio 2008 и 2010, Microsoft Expression Blend и другими для простой разработки и развертывания основанных на Silverlight, кросс-платформенных RIA. Хотя Silverlight действительно содержит общезыковую среду выполнения, у него нет абсолютно никакой зависимости от любой из версий .NET Framework. Бесплатный и небольшой по размеру (около 5,96 Мб для Windows и 8,71 Мб для Mac) плагин Silverlight имеет общезыковую среду выполнения компоненты библиотеку базовых классов. Если у пользователя не установлен плагин Silverlight, то автоматически будет предложено установить его для просмотра приложения Silverlight.

### Хронология Silverlight

В настоящее время доступны четыре версии Microsoft Silverlight: Silverlight 1, Silverlight 2, Silverlight 3, и Silverlight 4 [1,2].

**Silverlight 1.** Это выпуск Silverlight не имел общезыковой среды выполнения. Он поддерживал небольшое подмножество XAML и множество возможностей, которые предвещали будущее Silverlight. Возможно самый очевидный аспект Silverlight 1 – то, что приложения написаны или полностью на XAML или в смеси с JavaScript и объектной моделью документов, чтобы управлять интерфейсом пользователя. Так как нет общезыковой среды выполнения, нет фазы компиляции, и JavaScript интерпретируется на клиентской стороне.

**Silverlight 2.** Вскоре после того, как Silverlight 1 был выпущен предварительный релиз следующей версии Silverlight. Этот выпуск был известен как Silverlight 1.1, наиболее значительным аспектом которого является кроссплатформенная общезыковая среда выполнения. Хотя Silverlight 1 мог использоваться, чтобы разработать некоторые впечатляющие и «богатые» медиа-приложения, его возможности значительно расширились со способностью выполняться на платформе .NET. Самая большая недостающая возможность Silverlight 1.1 – это неточный набор стандартных элементов управления. Это делало разработку пользовательских интерфейсов трудной. Обработка входных событий была также трудной, так как события могли перехватываться только в корневом контейнере. Затем вручную приходилось распространять события к дочерним объектам. Использование фокуса ввода было также запутанным.

В марте 2008, Microsoft объявила, что Silverlight 1.1 будет фактически выпущен как Silverlight 2, так как набор его возможностей значительно расширился.

**Silverlight 3.** В марте 2009 Microsoft випустила бета-версію Silverlight 3. В юліе 2009 появилася среда розробки Expression Blend 3.

Silverlight 3 являється розширенням Silverlight 2 і головним образом забезпечує удосконалення можливостей графіки, управління медіа, області розробки програм (доповнітельні елементи управління, удосконалена підтримка зв'язування елементів управління з даними, підтримка внебраузерних програм), а також інтеграцію з Expression Blend 3.

### Silverlight 4

Microsoft головним образом фокусувалася на можливостях, управляємх медіа до версії Silverlight 3. Silverlight 3 представив ключеві можливості інтеграції даних для легкої розробки програм, управляємх даними. Однак, якщо необхідно розробити програму, управляєму даними, з функціональними можливостями, такими як друку, інтеграція документів, створення звітів, «багаті» офлайнні можливості і інтеграція з локальними пристроями, то надо починати з Silverlight 4 [1, 3].

Бета-версія Silverlight 4 була швидко випущена в ноябрі 2009, а остаточний реліз Silverlight 4 появилася 12 априля 2010 во время конференції DevConnection.

Архітектура програмної платформи Silverlight 4 приведена на рис. 2.

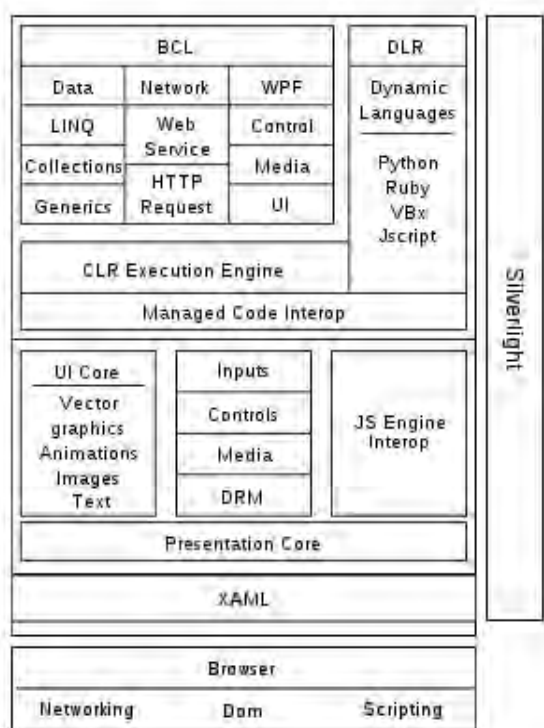


Рис. 2. Архітектура платформи Silverlight 4

Silverlight 4 має наступні ключеві особливості:

1. Можливість друку контенту з використанням нового API друку, дозволяє розробникам створювати користувацькі візуальні представлення друку, які можуть бути інтегровані з локальними принтерами або можуть бути збережені як файли (XPS, PDF).

2. Підтримка контекстних меню являється можливістю по умовчання, забезпечуючи інтерфейс користувача подібний існуючому в настільних програмах.

3. Підтримка *Drag and Drop* і операцій з буфером обміну дозволяють інтеграції локальних файлів або даних з RIA Silverlight.

4. *Нові елементи управління*, такі як RichTextBox, який дозволяє розробникам надавати область редагування тексту з підтримкою текстового форматування, гіперссылок і зображень в програмах Silverlight. Нові елементи управління WebBrowser і WebBrowserBrush дозволяють інтегрувати HTML-контент в програму, які є внебраузерними.

5. *Розширення можливостей існуючих елементів управління*, таких як DataGrid підтримкою сортування і зміни розмірів, а також операцій з буфером обміну.

6. *Удосконалена прив'язка до даних*

7. *Удосконалені можливості локалізації* додають підтримку для двонаправленого тексту, мов, на яких пишуть справа-наліво і доповнітельних 30 нових мов.

8. *Введення вікон повідомлень* для забезпечення основаних на панелі завдань традиційних повідомлень на клієнтській машині, удосконалює інтерфейс кінцевого користувача, забезпечуючи непротиворечивий підхід, як багато інших традиційних програм.

9. *Підтримка клавіатурного доступу в повноекранному режимі* і простої реалізації підтримки колісика миші забезпечує обробку подій MouseWheel і відповідно значительне удосконалення зручності і простоти використання програм.

10. *Можливість інтеграції з Web-камерою і мікрофоном* дозволяє розробляти інтерактивні корпоративні і клієнтські програми, які підтримують голосове взаємодія і проведення відео-конференцій.

Можливість для програм Silverlight бути "довіреними" так же як внебраузерні програми відкриває багато можливостей, які забезпечують традиційні WPF-програми, які можуть викликати «родний» код за межами «песочниці» на клієнтській машині:

1. *Підтримка пізнього зв'язування* з об'єктами, отриманими від HTML DOM або від Automation API. Це забезпечує можливість інтеграції з

СОМ-приложениями, такими как приложения Microsoft Office (например, Word, Excel, и Outlook) на Windows-клиентах и интеграцию других подключенных устройств.

2. Доступ к Папкам «Мои...» из приложения позволяет доступ ко всем папкам «Мои документы», «Мои видеозаписи», «Изображения», и «Музыка» для Windows и Mac и читать и записывать файлы из приложения, работающего в среде «песочницы».

3. Легкое развертывание и управление, избавляющее от необходимости использования междомеинных файлов политик доступа (ClientAccess.xml или CrossDomainAccess.xml), а также возможность создания групповых политик для управления доверенными приложениями.

Silverlight 4 действительно включает несколько ключевых расширений, чтобы защитить, обработать и доставить более дружественные с медиа RIA:

1. Поддержка офлайн-управления цифровыми правами поможет доставить медиа в офлайн-режиме в более защищенной и управляемой среде.

2. Как упоминалось ранее, возможности интеграции с Web-камерой и микрофоном и возможность записи аудио и видео клиентской стороны, делающая, обеспечивает многочисленные возможности для разработки интерактивных аудио/видео RIA.

Следующие усовершенствования основной платформы и компонентов поможет улучшить интерфейс конечного пользователя и разрабатывать корпоративные приложения в режиме быстрой разработки.

1. Официальная поддержка web-браузера Google Chrome - это вероятно конечный шаг к провозглашению Silverlight кроссбраузерной платформой и конечно осчастливит фанатов Google.

2. Введение поддержки широковеб-сетевых взаимодействий по протоколу UDP поможет улучшить потребительские свойства и стабильность приложений, более эффективно используя сетевые ресурсы.

3. WCF-службы RIA помогают разрабатывать многозвенные приложения в режиме быстрой разработки.

4. Полная оптимизация производительности улучшает производительность запуск и выполнения приложений Silverlight 4 по сравнению с приложениями Silverlight 3.

5. Visual Studio 2010 позволяет разработку пользовательского интерфейса для приложений Silverlight 4 (и Silverlight 3) и вводит улучшенную привязку данных и интеграцию WCF-служб RIA, а также другие расширения, чтобы улучшить опыт разработки.

6. Silverlight 4 и инструментальные средства разработки Windows Phone допускают разработку интерактивных приложений Silverlight для мобильного телефонов серии Windows 7. Вы можете разрабатывать высококачественные медиа-приложения (включая интеграцию с видео камерой и микрофоном) для игровых приложений для мобильных телефонов Windows.

## Выводы

Silverlight – новая технология от Microsoft, предназначенная для разработки насыщенных Web – приложений. Особенности версий Silverlight 4 – интенсивное использование графики, анимации, работа с медиа-файлами, а так же эффективное взаимодействие с данными и серверными компонентами.

## Список литературы

1. Байдачный С.С. Silverlight 4: Создание насыщенных Web-приложений / С.С. Байдачный. – М.: СОЛОН-ПРЕСС, 2010. – 288 с.

2. Центр разработки Silverlight [Электронный ресурс]. – Режим доступа к ресурсу: <http://msdn.microsoft.com/ru-ru/>.

3. Сообщество пользователей Silverlight [Электронный ресурс]. – Режим доступа к ресурсу: <http://silverlighter.ru/>.

Поступила в редколлегию 20.09.2010

Рецензент: д-р техн. наук, проф. И.В. Рубан, Харьковский университет Воздушных Сил им. И. Кожедуба, Харьков. .

## АНАЛІЗ ОСНОВНИХ ОСОБЛИВОСТЕЙ SILVERLIGHT 4

Ю.Е. Парфьонов, В.М. Федорченко, О.В. Щербаков

В статті розглянуто основні принципи побудови технологічної платформи від Microsoft – Silverlight 4. Проаналізовано середовища, які існують для розробки кроссплатформених додатків. Розглянуто історію розвитку платформи Silverlight. Наведено структуру і можливості Silverlight 4 щодо розробки інтерактивних інтернет-додатків нового покоління.

**Ключові слова:** Silverlight, RIA, WPF, XAML, Web 2.0, платформа, інтерфейс користувача, WCF, інструментальні засоби розробки ПЗ.

## ANALYSIS OF FUNDAMENTAL SPECIFICS SILVERLIGHT 4

Y.E. Parfyonov, V.N. Fedorchenko, A.V. Shcherbakov

In article main principles on-structure technological platforms from Microsoft – Silverlight 4 are considered. Existing environments for working out cross-platform appendices are analysed. The history of development of platform Silverlight is considered. The structure and possibilities Silverlight 4 on working out of interactive Internet appendices of new generation is stated.

**Keywords:** Silverlight, RIA, WPF, XAML, Web 2.0, framework, user interface, WCF, development tools software.