



RIA – Rich Internet Applications

«Обогащенные Интернет приложения»



Коротко о докладчике

- 10 лет опыт работы в IT
- 4 года опыт разработки RIA приложений, в качестве архитектора.
- Текущий проект 1,5 года разработки, школьная ERP, более 10Мб сгенерированного сжатого JavaScript кода, >5000 классов, 15Мб исходного кода.



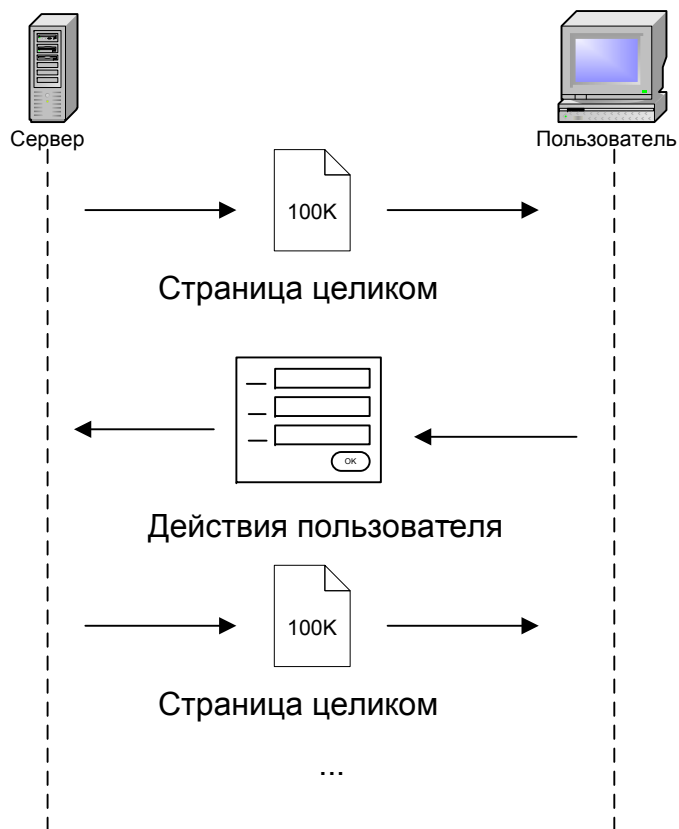
Что такое RIA?

Rich Internet application (RIA, «богатое Интернет-приложение») — это приложение, доступное через [Интернет](#), богатое функциональностью традиционных настольных приложений, не поддерживаемой браузерами непосредственно. Как правило, приложение RIA:

- передаёт веб-клиенту код пользовательского интерфейса, обмениваясь с сервером данными по определенному протоколу;
- запускается в браузере и не требует дополнительной установки ПО;
- запускается локально в среде безопасности, называемой «[песочница](#)» (sandbox).



Классическое HTML приложение



ria_demo_2009@mail.ru



RIA приложение



ria.demo.2009@gmail.com



Немного цифр

	mail.ru	gmail.com
Начальная страница	24 запроса 116 Кб	33 запроса 379 Кб
Открыть ПИСЬМО	17 запросов 71 Кб	3 запроса 2 Кб



Демонстрация

- mail.ru
- gmal.com

- Google.docs
- Google.wave



Преимущества RIA приложений для пользователя, по сравнению с классическими web приложениями

- Меньше сетевой трафик
- Быстрее отклик сервера, приложение более интерактивное, например проверка правильности ввода, без запроса на сервер.
- Обмен данными с сервером не обновляет страницу целиком
- Асинхронная передача данных, можно работать, пока приложение обменивается данными с сервером
- Меньше нагрузка на сервер
- Более отказоустойчивое
- Более удобный, функциональный и красивый интерфейс пользователя
- Drag & Drop
- Более плотное взаимодействие с компьютером пользователя, например доступ к локальному диску или сканеру и т.п.
- Частичная возможность работы в режиме offline



Недостатки

- Дороже разработка
- Часть данных «невидима» для поисковых систем
- По сравнению с классическими настольными приложениями – зависимость от подключения к Интернету и ограничения песочницы.



История вопроса

- В хронологическом порядке:
 - JAVA
 - *ActiveX*
 - AJAX
 - Adobe Flex
 - Microsoft Silverlight
- Последние веяния:
 - Google Gears
 - HTML 5.0



Java

История

- 1990 – начало разработки
- 1995 – первый релиз
- 1997 – судебный процесс с Микрософт
- 1998 – Java2 – не поставляется с Windows и требует отдельной загрузки
- 2006 – Java SE 6
- 2008 - JavaFX
- 2009 – покупка Sun компанией Oracle



Java

Текущее состояние

- Одна из наиболее популярных платформ для серверных приложений
- Фактически не используется для desktop. UI библиотека Swing крайне не популярна
- Sun выпустила JavaFX для упрощения разработки для web и desktop.
- Судьба Sun не ясна, Java точно останется, но в какую сторону будет ее развивать Oracle не понятно.
- В 2010 запланирован релиз Java SE7, ничего принципиально нового там пока не планируется, релиз несколько раз откладывался.



Java

Преимущества

- Наиболее полнофункциональная платформа.
 - Можно подключать оборудование устанавливая драйвера, загружать и использовать Native библиотеки (e.g. VPN via browser)
 - Можно перетаскивать приложения со страницы браузера на desktop
- Громадное сообщество разработчиков
- Множество готовых библиотек и фреймворков на все случаи жизни.
- Высокая производительность (на 7-9% ниже чем у native кода).

Недостатки

- Не мудрая маркетинговая политика фирмы Sun
- Как стандарт UI используется библиотека Swing – ресурсоемкая и очень сложная в освоении.



ActiveX

Как технология для RIA обладает рядом серьезных недостатков

- Платформа только IE + Windows
- Нетривиален в разработке и отладке
- Большой период освоения технологии
- Было множество проблем с безопасностью

Пожалуй единственное бесспорное достоинство

- Исполняется native код



AJAX

История

- Первые приложения в 1997 (IE3 IFRAME), 1998 MS Remote Scripting
- 1 апреля 2004 – запуск Gmail.
- Термин AJAX официально использован в 2005
- GWT 1.0 – 2006



AJAX

Текущее состояние:

- Самая популярная платформа для RIA
- Множество различные фреймворков и библиотек
- Фактически любой крупный сайт содержит немного AJAX



AJAX

Преимущества

- Есть в любом браузере
- Прост в освоении
- Быстрый цикл разработки
- Развитый инструментарий
- Развитое сообщество

Недостатки

- Кроссбраузерная несовместимость
- Сложно/дорого разрабатывать большие приложения
- Сложно отлаживать, дорого поддерживать
- Низкая производительность



Adobe Flex

История

- Формально Flex начался с Flash, который в свою очередь начался в 1986 году с технологии векторного морфинга
- Macromedia Flash player 2 Июнь 1997
- Adobe Flex SDK позиционируется как платформа для RIA с 2006 года.
- Февраль 2005 Adobe Flex 3
- Adobe Flex 4.0 релиз планируется на 2010



Adobe Flex

Текущее состояние

- Самая распространенная среда исполнения на текущий момент.
- Громадное кол-во UI компонентов и библиотек
- Большое community, в основном сосредоточенное на визуализации
- Наиболее зрелая платформа
- Бесспорно самая красивая платформа
- Adobe очень активно продвигает Flex в качестве платформы для RIA
- Как опция специальный очень эффективный протокол обмена между клиентом и сервером (AMF), во многом снимает проблемы использования http для RIA.
- Активно развивается, пользуется популярностью в Java community



Adobe Flex

Преимущества

- Быстрый цикл освоения и разработки
- Развитое community
- Много учебных материалов и документации
- Интеграция с дизайнерскими инструментами от Adobe
- Надежная простая и эффективная платформа

Недостатки

- Предвзвешенное мнение «Flex это для мультимедиа», несмотря на надуманность и не серьезность одна из очень серьезных проблем.
- Существенные (по сравнению с Java) ограничения песочницы
- Многие разработчики акцентированы на визуализации, как следствие плохой серверный код.



Microsoft Silverlight

История

- Бета – декабрь 2006
- Релиз 1.0 – апрель 2007
- Релиз 3.0 – июль 2009

- Ограниченная поддержка браузерами.



Microsoft Silverlight

Текущее состояние

- Наиболее динамично развивающаяся платформа RIA
- В качестве языка используется любой язык платформы .Net
- Агрессивный маркетинг Microsoft
- Платформа по умолчанию устанавливается на Vista & Windows 7
- Активно набирает популярность среди .Net разработчиков
- Использует инструментарий .Net
- Платформа разработки фактически ограничена Windows (есть ограниченная поддержка для Linux)
- Идеологически много общего с Flex



Microsoft Silverlight

Преимущества

- Общий с .Net язык и среда разработки
- Как следствие для Silverlight открыто все сообщество .Net разработчиков
- Активно развивается и продвигается Microsoft
- Автоматически доступен в новых инсталляциях Windows

Недостатки

- Доступен не для всех комбинаций платформа/браузер
- IDE – Visual Studio – по функционалу проигрывает всем остальным платформам, дорогая.
- Все еще мало распространен, мало известен
- Мало библиотек и компонентов, традиционно бедный для MS технологий набор open source компонент (по сравнению с другими платформами).



Google Gears

Модуль расширения для браузера, позволяет AJAX приложениям расширить ограничения песочницы

- Доступность ресурсов offline
- Локальная SQL Lite база данных, синхронизация данных с сервером
- Асинхронные задачи
- Выбор нескольких файлов для загрузки
- Превью картинок
- Drag & Drop в браузер.
- ...



HTML 5.0

- Рисование 2D-картинок в реальном времени;
- Контроль над проигрыванием медиафайлов;
- Хранение данных в браузере;
- Drag-and-drop;
- Работа с сетью, push технология

Основные решаемые проблемы

- Работа с большими DOM моделями
- Передача данных по инициативе сервера
- Работа с медиа файлами.

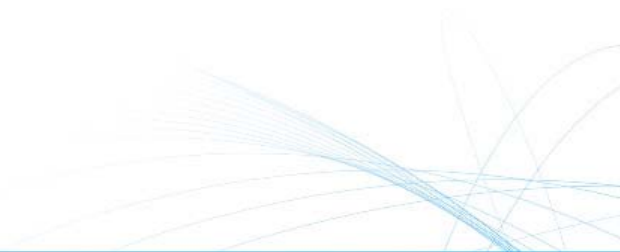


Современные тенденции и проблемы

- Проблемы:
 - Отсутствие стандарта
 - Как следствие разобщенность сообщества разработчиков
 - Высокий порог вхождения
 - Не зрелось инструментария
 - Множество мифов и заблуждений
- Тенденции
 - Рост популярности RIA, особенно на скрытом корпоративном рынке
 - Сближение платформ
 - Производители браузеров выбирают AJAX
 - JavaScript JIT
 - CSS 3D (<http://www.w3.org/TR/css3-3d-transforms/>)



Вопросы





Если осталось время несколько демо

