

Протоколы передачи данных для систем дистанционного обучения

Гроцев Алексей Романович
аспирант, кафедра технологий разработки программного обеспечения
Рижский Технический университет, ул. Межа, 1/3, г. Рига, LV-1048, (371)67089571,
Латвия
root@smallserver.org

Аннотация

В статье рассматриваются протоколы передачи данных для использования в RIA-приложениях. Данные протоколы могут быть использованы при передаче больших объемов данных, что свойственно дистанционному обучению: презентации, мультимедийные приложения и т.п. Приводится краткое описание современных возможностей приложений, их особенности и потребности, а так же описание и сравнение некоторых реально используемых протоколов.

This article studies the possible protocols for data transmission to fit RIA needs as well as the solutions to the problem of data transmission for remote education systems. The RIA possibilities are described including their special needs and features. The main part of article describes modern protocol structure and some use cases.

Ключевые слова

Передача данных, AMF, RTMP, plain-text, протоколы, RIA
Data transmission, AMF, RTMP, plain-text, protocols, RIA

Введение

Проблема повышения эффективности и качества образования в настоящее время является актуальной, т.к. спрос на высококвалифицированных специалистов и количество необходимой им информации неуклонно растет, а значительно увеличить время обучения не представляется возможным. Поэтому в последние годы широкое распространение получило компьютерное обучение на базе Интернета, используемое при дистанционном образовании и повышении квалификации. Для поддержки компьютерного обучения создаются и широко используются различные среды, платформы и порталы, обучающие и тестирующие программы, исследуются и разрабатываются методы и технологии электронного обучения, а так же специализированные алгоритмы управления процессом обучения и взаимодействия с обучаемым [1-4]. Однако, техническая сторона вопроса, а именно эффективная передача больших объёмов данных, еще недостаточно исследована. В основном учебные материалы содержат мультимедиа-данные, передаваемые по сети клиенту, который их выводит на экран или проектор.

Rich Internet application (богатое интернет-приложение) — это приложение, доступное через интернет, богатое функциональностью традиционных настольных приложений, не поддерживаемой браузерами непосредственно и не требующее скачивания дополнительных файлов на компьютер пользователя. На сегодняшний день доступны следующие варианты технологий построения RIA:

- Adobe Flash;
- Adobe Flex;
- Java Applets;
- Microsoft Silverlight.

Термин «RIA» впервые был упомянут компанией Macromedia в официальном сообщении от марта 2002 года. Эта концепция существовала несколькими годами ранее со следующими названиями:

- Remote Scripting, Microsoft, около 1998 года;
- X Internet, Forrester Research в октябре 2000 года;
- Rich (web) client;
- Rich web application.

Google в конце 2009 года анонсировала платформу для совместной работы и общения в реальном времени Google Wave. Однако, по заверениям разработчиков, она построена исключительно на основе возможностей HTML5, т.е. будет запускаться в последних версиях браузеров без дополнительных надстроек, что не даёт возможности назвать эту технологию полноценным RIA.

Отличительные черты передачи данных между обычными программами и RIA состоят в том, что обычно интернет-приложения требуют постоянного (или очень частого) обмена данными с сервером, чтобы отражать последнюю актуальную версию данных, например меняющиеся слайды презентации и комментарии к ним, видеопоток с камер преподавателя и студентов и т.д. Для этого необходим протокол, позволяющий передавать данные с минимальными задержками на парсинг, в идеальном случае быть бинарным с возможностью сериализации данных. Целью данной статьи является сравнение протоколов ARM, RTMP, XML и plain-text и определение наиболее подходящего для передачи данных в системах дистанционного обучения.

Описание протоколов

AMF (Action Message Format) – сравнительно новый протокол обмена сообщениями, разработанный Adobe. Его спецификация была опубликована в декабре 2007 года и он предназначен для передачи бинарных данных между flash-приложением, написанным на ActionScript и сервером, в виде которого может быть как сервер от разработчиков, Flash Media Server (на данный момент это Flash Media Interactive Server), так и сторонние разработки, такие как Wowza/Red5 и другие. Перед посылкой данные сериализуются, а все строки приводятся в UTF-8. Стандарт AMF3 даёт возможность передавать 13 разных типов данных [5], которые перечислены в таблице ниже:

Таблица 1

Типы передаваемых данных в AMF3

Тип	Описание
undefined-type	Тип неизвестен, данные не передаются
null-type	Хотя значение данных - пустота, сам объект существует
False-type	Логическое ЛОЖЬ
true-type	Логическое ПРАВДА
integer-type	Числа без знака в интервале от 0 до 2^{28}
double-type	Числа со знаком в интервале от 0 до 2^{28}
string-type	Описание цепочки символов в кодировке Unicode

xml-doc-type	Описание класса flash.xml.XMLDocument, размер до 256MB
Date-time	Время в формате Unix timestamp UTC+0
Array-type	Определение массива. Массивы могут быть нумерованные, ассоциативные и пустые
object-type	Самый частоиспользуемый тип. Содержит объект, его параметры и указатели на них.
xml-type	Необходим для поддержки формата E3X, схож с типом xml-doc-type
bytearray-type	Отдельный тип данных для бинарных цепочек, размер до 256MB

Чтобы передать данные с помощью протокола AMR, необходимо создать инстанцию объекта NetConnection. После подключения можно использовать передачу используя в виде запаковщика данных встроенные во Flash/Flex классы:

```
NetConnection.defaultObjectEncoding =
flash.net.ObjectEncoding.AMF0;
SharedObject.defaultObjectEncoding =
flash.net.ObjectEncoding.AMF0;
private var echoResponder:Responder = new Responder(echoResult,
echoStatus);
nc.call("remote_Procedure", echoResponder, "Hello world");
```

В данном примере на сервер методу remote_Procedure была передана строка Hello world. Если метод вернёт результат, будет вызван обратный (callback) метод – echoResponder.

Похожий на предыдущий, но имеющий расширенные возможности, протокол **RTMP** (Real-Time Messaging Protocol) был так же придуман Adobe, и его спецификация появилась в свободном доступе только в середине 2009 года. Протокол задумывался как средство быстрого начала общения, т.к. изначально он мог передавать данные только с Flash-клиентов, которые, по статистике на декабрь 2009 года, установлены на 99% всех компьютеров [6], и в виде источника видео может служить любая малобюджетная видеокамера. Так как протокол позволяет передавать видео, его структура отличается от AMF, однако, содержит в себе тело AMF-объекта. Описание типов данных RTMP-протокола приведено во второй таблице.

Таблица 2

Типы передаваемых данных в RTMP

Тип	Описание
Chunk size	Размер "куска" данных
Bytes Read	Количество переданных данных (для контрольной суммы)
Ping	Контрольная единица потока, содержит внутренний объект
Server BW	Скорость скачки с сервера (статистика)
Client BW	Скорость скачки от клиента (статистика)
Audio Data	Пакет с аудио-данными
Video Data	Пакет с видео-данными
Flex Stream	Поток с внутренними объектами
Flex Shared Object	Поток с общедоступным объектом с информацией о размере
Flex Message	Статус-сообщение с возможными аргументами
Notify	Вызов, для которого не предусмотрен callback
Shared Object	Общедоступный объект с подтипами
Invoke	Данные вызова метода без возвращаемого значения, для внутреннего использования

Object	Указывает один или несколько FLV-тегов, каждый тег содержит 11 байтов заголовков, (1 байт – тип, 3 байта – размер, 3 байта – время, 1 байт, – уточнение времени, 3 байта – идентификатор потока), после которого идёт тело FLV-кадра, который заканчивается 4 байтами-меткой, которая так же содержит размер тела.
--------	--

Вызов методов и передача аргументов производится аналогично AMF.

Расширяемый язык разметки (**XML**, eXtensible Markup Language), также может быть использован для синхронизации данных с сервером. Язык разметки представляет подтип языка SGML, который понятен как компьютеру, так и человеку. Данными в протоколе передачи, использующем XML, могут быть любые символы, которые не включают в себя „<” и „>”. Все данные передаются в таком виде, в каком они были запакованы в XML, т.е. передача UTF-8 текста должна быть реализована именно на транспортном уровне, а не на уровне самого формата, т.е. следующий корректный XML-код должен быть передан без ошибок, хотя и содержит как русский текст, так и китайские иероглифы из расширенного набора Unicode.

```
<?xml version="1.0" encoding="UTF-8"?>
<俄語>Китайский язык</俄語>
```

Бинарные данные, обычно, не передают в "чистом виде", а кодируют используя Base64. XML широко распространён среди интернет-приложений благодаря своей реализации в протоколе XML-RPC, который позволяет запрашивать данные с сервера вызывая удалённые процедуры и получая данные в виде ответов. Такие приложения, как GoogleTalk и Jabber тоже используют XML для связи со своими клиентами [7, 8].

Конечно, существуют и неструктурированные форматы данных, передаваемых текстом без форматирования (plain-text) как в XML, допустим это протоколы IRC, HTTP, SMTP и многие другие. В них данные передаются такими, какие они есть, обычно отделяя системную информацию (заголовки) от содержимого (тела) двойным переносом строки. Обычно такого рода протоколы уникальны в пределах своей задачи и не могут служить для решения разнообразных задач, как AMR или XML.

Заключение

Оценивая возможности перечисленных протоколов, а так же опираясь на схожие исследования в данной сфере [9-11] можно сделать следующие выводы:

- Простота внедрения зависит, в основном, от доступности библиотек/расширений языка программирования, где уже реализован необходимый протокол. Например, XML парсеры есть практически во всех современных языках программирования, тем не менее для RTMP библиотеки доступны только на C/Java/PHP/Ruby/C#/Erlang.
- Необходимые вложения на внедрения - если не считать уже рассмотренной простоты внедрения, это, безусловно, лёгкость разработки и поддержка средствами профилирования/отладки. На данный момент только XML обладает широкой базой модулей интеграции в различные среды разработки. Однако, программу, использующую любой другой текстовый протокол, удобнее отлаживать, чем бинарно-ориентированную в связи с возможностью сделать полный сброс пришедших данных в отладчик для более подробного анализа без использования дополнительных инструментов декодинга.

- Если сравнивать разнообразие передаваемых типов, то тут лидирует RTMP, т.к. только у него из рассматриваемых протоколов есть возможность передачи аудио- и видео-данных.
- Как было сказано, RIA приложения чувствительны к перебоям с доставкой данных, соответственно, протокол должен быть оптимизирован для низкоскоростных линий и быть устойчивым к разного рода воздействиям, таким, как потери и задержки. RTMP и AMF, являясь бинарными протоколами, более предпочтительны [12], нежели текстовые протоколы из-за хорошего сжатия и возможности асинхронной передачи данных, что в случае с текстовыми протоколами могло привести к каше на стороне приёмника.

Таким образом, для систем дистанционного обучения в качестве удобного и выгодного применения можно практически однозначно рекомендовать следующие протоколы передачи данных: при использовании мультимедиа-данных (моделирование, демонстрация физических процессов и т.п.) – протокол RTMP; в остальных случаях – протокол AMF.

Литература

1. Башмаков А.И., Башмаков И.А. Разработка компьютерных учебников и обучающих систем. – М: Информационно-издательский дом «Филинь», 2003. – 616 с.
2. Галеев И.Х. Модель управления процессом обучения в ИОС // Международный электронный журнал "Образовательные технологии и общество (Educational Technology & Society)" - 2010. - V.13. - №3. - с.285-292. - ISSN 1436-4522. URL: <http://ifets.ieee.org/russian/periodical/journal.html>
3. Зайцева Л.В. Технология разработки адаптивных электронных учебных курсов для компьютерных систем обучения // Международный электронный журнал "Образовательные технологии и общество (Educational Technology & Society)" – 2008. – V.11. – № 1. – с. 400 – 412. - ISSN 1436-4522. URL: <http://ifets.ieee.org/russian/periodical/journal.html>
4. Kristensen T., Lamo Y., Mughal K., Tekle K.M., Bottu A.K. Towards a Dynamic, Content based e-Learning Platform // Proceedings of the 10th IASTED International Conference on Computers and Advanced Technology in Education (CATE 2007). October, 8-10, 2007, Beijing, China, – p. 107 – 114.
5. Action Message Format -- AMF 3 / Интернет – http://download.macromedia.com/pub/labs/amf/amf3_spec_121207.pdf (дата обращения: 20.01.2011)
6. Adobe - Flash Player Statistics / Интернет – http://www.adobe.com/products/player_census/flashplayer/ (дата обращения: 20.01.2011)
7. Cover Pages: Jabber XML Protocol / Интернет – <http://xml.coverpages.org/jabber.html> (дата обращения: 20.01.2011)
8. Open Communications - Google Code / Интернет – http://code.google.com/apis/talk/open_communications.html (дата обращения: 20.01.2011)
9. Гроцев А. Передача видео используя RTMP-протокол // Сб. научн. трудов 49-ой научн.-техн. конф. РТУ. – Рига: Риж. техн. ун-т, 2008. – с. 86. – 91. (на латышском языке)

10. Bodendorf F., Gotzelt K.-U., Schertler M., Soy M., "Process Design for e-Teaching with Streaming Media", Advanced Technology for Learning, Vol. 2, No. 2, 2005 – pp. 66-75.
11. Grocevs A., Prokofjeva N. Modern Internet Technology Analysis // Scientific proceedings of Riga Technical University. Computer Science. Applied Computer Systems.– 5th edition. – 2008. – Vol. 17. – pp. 103. - 108.
12. Flex 3 - Using RTMP / Интернет – http://livedocs.adobe.com/flex/3/html/security2_11.html (дата обращения: 20.01.2011)