Введение в предметную область, протокол HTTP

Стандартные протоколы

* HTML – Язык разметки гипертекста
* HTTP – предназначен для передачи гипертекста между клиентом и сервером

Протокол HTTP

* Протокол прикладного уровня
* Основа – клиент сервер
* Основной объект работы – ресурс, на который указывает URI. Ресурс – все что угодно.
* Обмен информацией – вопрос ответ
* Stateless протокол – отсутствие состояния. Т.е. сервер не запоминает то что один и тот же клиент обратился к нему с запросом, он всех обслуживает независимо, не запоминая сессий. Плюсы – масштабируемость, минус – состояние не сохраняется.

URI (identifier) – уникальный идентификатор ресурса, символьная строка позволяющая идентифицировать ресурс

URL (locator) – URI, позволяющий определить местоположение ресурса

URN (name) – URI содержащий единообразное имя ресурса (без местоположения)

REST

Подход к архитектуре сетевых протоколов, обеспечивающих доступ к информационным ресурсам.

Основные принципы

* Данные должны передаваться в стандартном формате – JSON, XML, HTML
* Сетевой протокол должен поддерживать кэширование, но не должен зависеть от сети, не должен сохранять состояние

Те REST приложение компонуется из набора ресурсов, каждый из которых имеет свой URI, URL.

Другой подход – RPC основан на вызове удаленных процедур. Набор процедур, которые находятся на одном уровне URL. RPC – сложная логика – информационные системы. REST – более простой – публичные сайты.

Запрос HTTP состоит из 3х частей.

1. Стартовая строка: метод URI HTTP версия – get /spip.html HTTP/1.1
2. Заголовки - массив элементов в формате ключ значение, содержат метаинформацию для сервера о запросе: Host : cs.ifmo.ru Accept : text/html Connection : close
3. Тело сообщения – набор данных, которые мы передаем на сервер.

Методы HTTP

* OPTIONS – определение возможностей сервера
* GET – запрос на ресурс
* HEAD – аналог GET, в котором отсутствует тело
* POST – передача данных ресурсу
* PUT – загрузка содержимого запроса на указанный URI

Коды состояния

* Состоят из 3х цифр
* Первая цифра – класс состояния
  + 1 - информационный
  + 2 - успешно
  + 3 - перенаправление
  + 4 – ошибка клиента
  + 5 – ошибка сервера
* Пример 404

Заголовки HTTP

Формат – ключ: значение

* General Headers – могут быть добавлены в любое сообщение
* Request Headers – используются только в запросах клиента
* Response Headers – используются только в запросах сервера
* Entity Headers – сопровождают любую сущность сообщения