

# Ajaxifying Legacy Web Applications

Anas Mughal

[anas.mughal@bluenog.com](mailto:anas.mughal@bluenog.com)

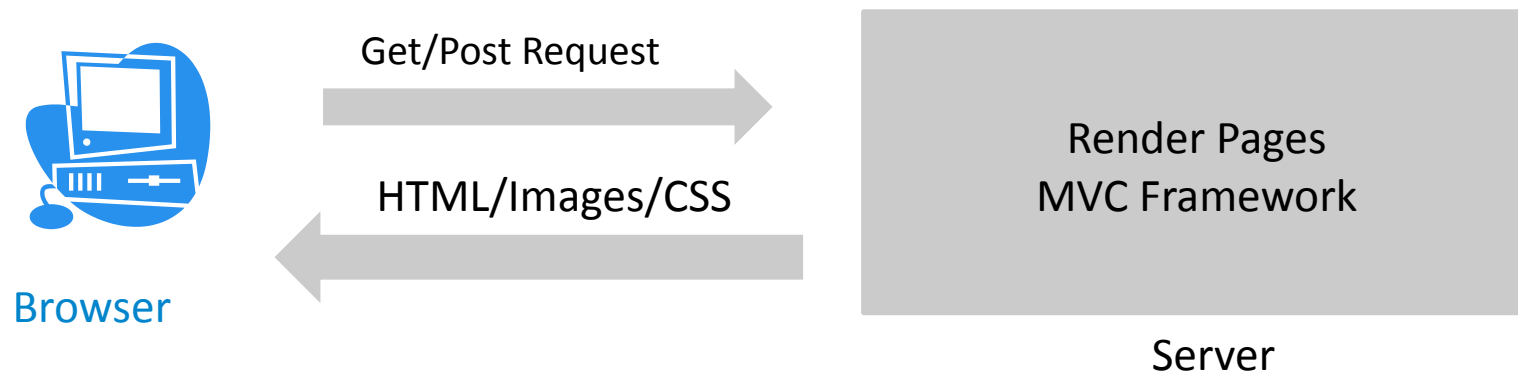


# Topics

- Overview: Legacy vs RIA
- Data Formats
- Tools
- Prerequisites
- Design Approach
- Demo
- QA

# Web 1.0 Applications

- Pages are rendered server-side
- Client-side is stateless
- Server-side MVC
- Full-page refresh



# Ajax Applications

- Stateful client
- Client-side session
- Server-side provides data services
- Client-side MVC framework
- No full-page refresh



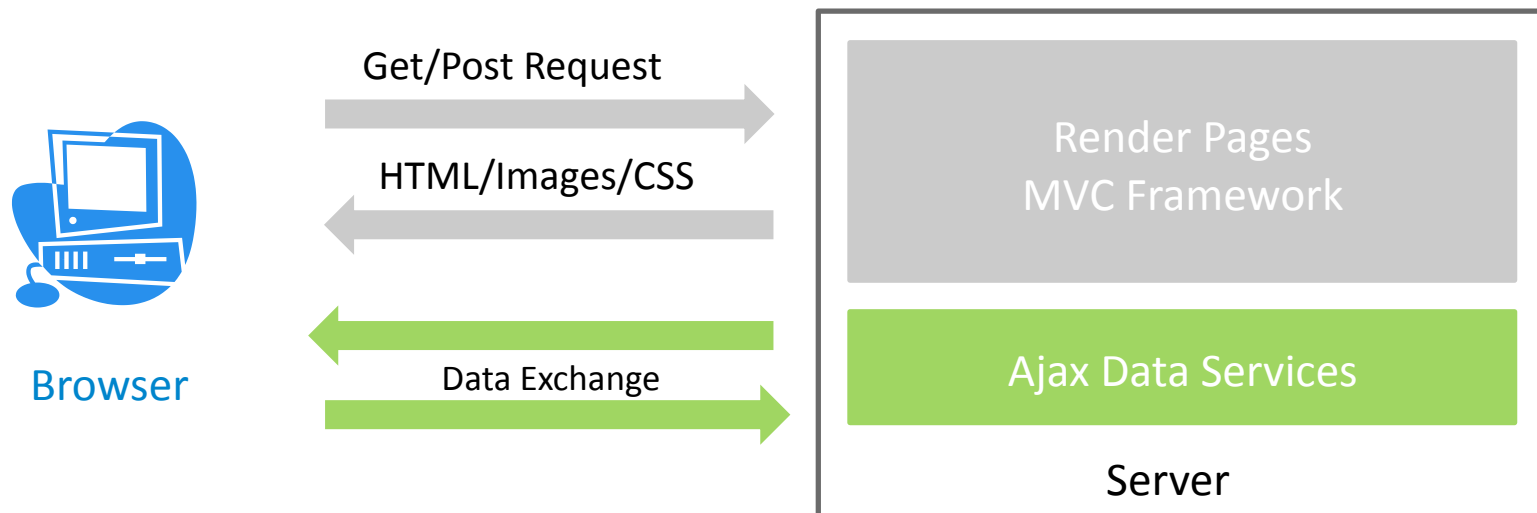
Browser



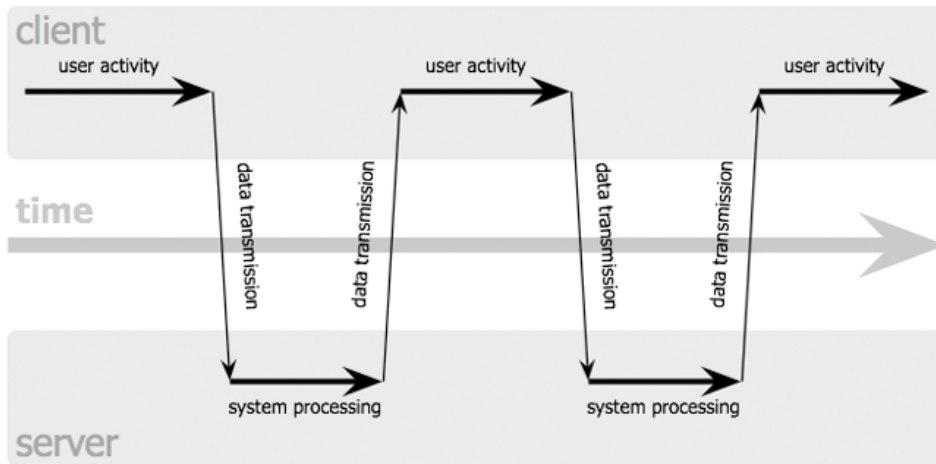
Server

# Suggested Approach

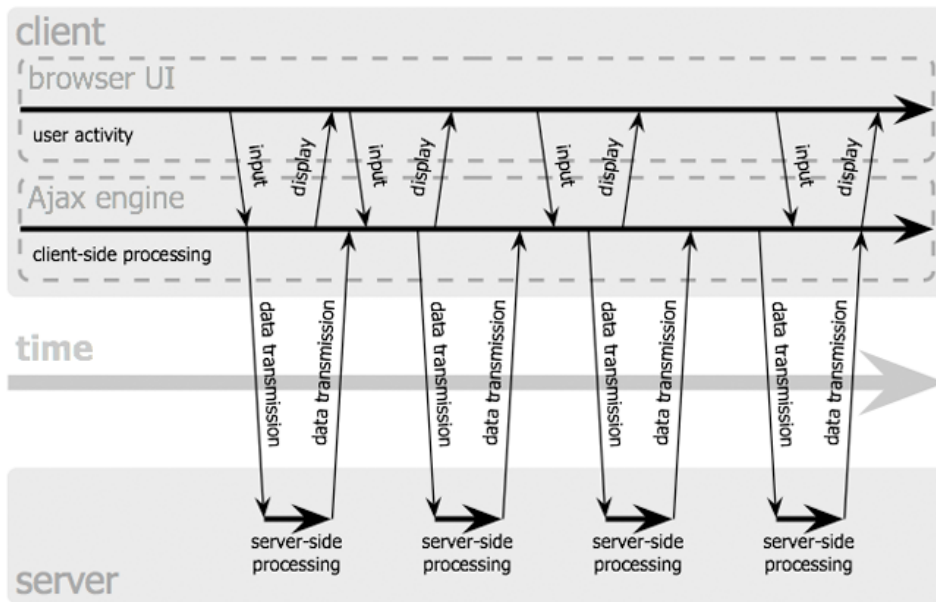
- Pages are rendered server-side
- Server-side MVC
- Ajax **controls/grids/forms** on client-side
- Exchange data with server-side



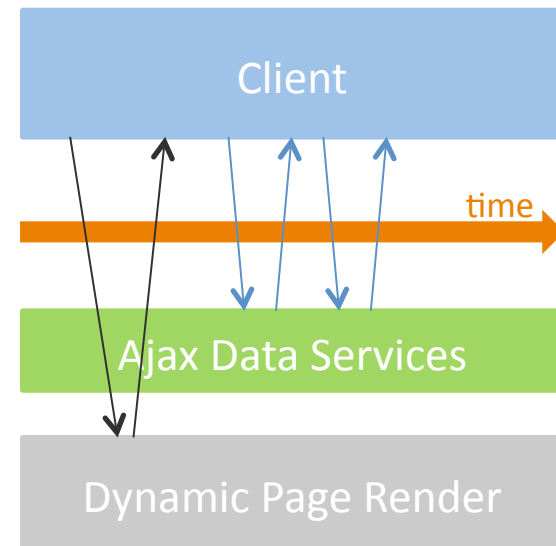
## classic web application model (synchronous)



## Ajax web application model (asynchronous)



Jesse James Garrett / adaptivepath.com



*Suggested Model*

# Data Formats

- JSON (JavaScript Object Notation)

- Native JavaScript structure.
- Lightweight data-interchange format
- Retains type information

```
{"names": ["Anna Maria", "Fitzwilliam", "Maurice"], "count": 3 }
```

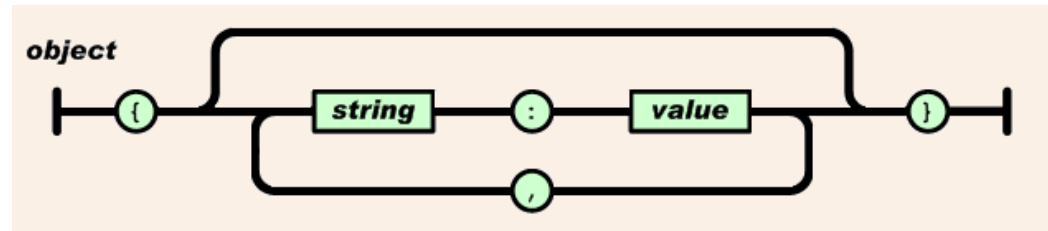
- XML

- Suited for data transformation needs.
- Parsing is CPU intensive.
- DOM may require large memory on client-side.
- Does not retain type information.

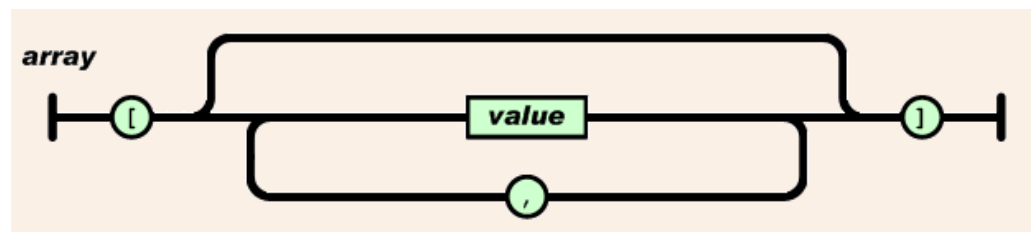
```
<xmldata>  
  <names>  
    <name>Anna Maria</name>  
    <name>Fitzwilliam</name>  
    <name>Maurice</name>  
  </names>  
  <count>3</count>  
</xmldata>
```

# JSON

- JSON is built on two structures:
  - A collection of name/value pairs.



- An ordered list of values.





# JSON Support in Frameworks

- Spring MVC Framework
  - Spring-json View (<http://spring-json.sourceforge.net>)
    - Set data in the model and pass control to the spring-json view
    - Supports: SOJO and JSON-lib
- Struts Framework
  - Struts 2 JSON Plugin (<http://tinyurl.com/b87ndu>)
    - Serializes entire action class variables
    - Provides incoming request interceptor
    - Example: <http://localhost:9090/struts2json/>
- Build your own
  - Convert data structures using appropriate JSON library
  - Set content-type to “application/json”

# Tools

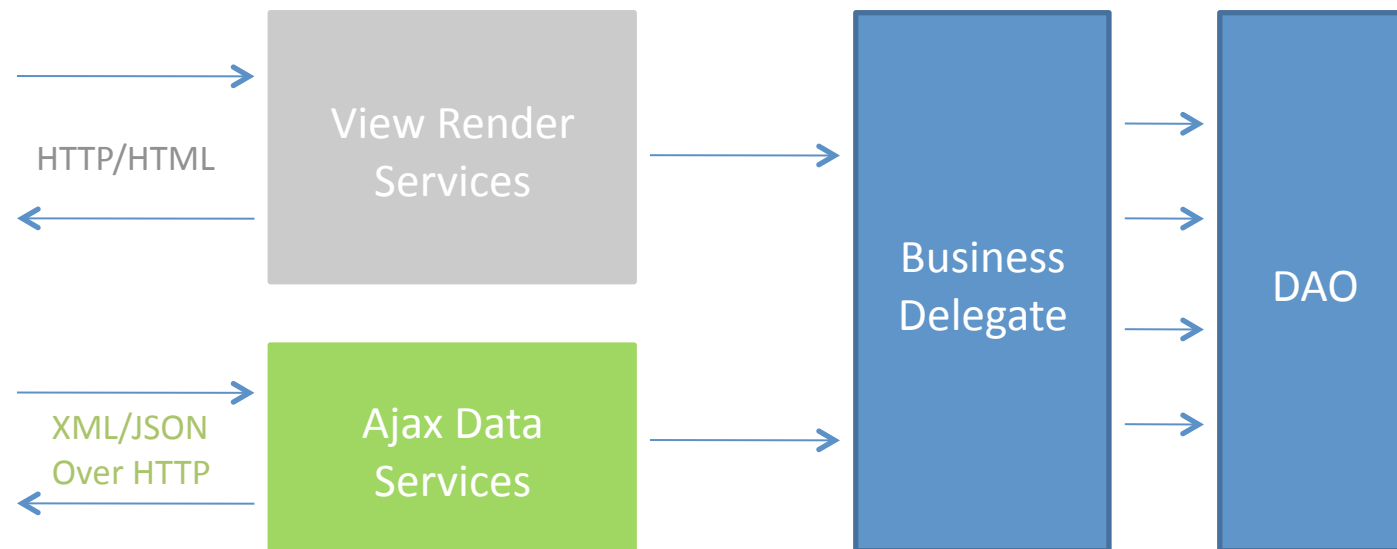
- JSON
  - Serializer: SOJO, json-lib (For other libs: <http://json.org>)
  - Formatter: <http://jsonformatter.curiousconcept.com/>
  - Visualizer: <http://chris.photobooks.com/json/default.htm>
  - Validator: [www.jsonlint.com](http://www.jsonlint.com) , [www.jshint.com](http://www.jshint.com)
  - Editor, Minifier, Formatter, Tree View: [jsoneditor.appspot.com](http://jsoneditor.appspot.com)
- Javascript Editor
  - SPKet Eclipse Plugin (www.spket.com)
- Javascript Debugger: Firebug
- HTML Validator

# Prerequisites

- Clean your HTML -- HTML Validator
- Select appropriate JSON library: (<http://json.org>)
  - SOJO
  - JSON-LIB
- Select appropriate Ajax framework:
  - ExtJS
  - Yahoo UI
  - GWT

# Design Approach

- Incorporate Ajax data services layer
- Integrate Ajax layer with business delegates



# Demo & Code Walk-thru

# Q & A

Anas Mughal

[anas.mughal@bluenog.com](mailto:anas.mughal@bluenog.com)