SYNERGY • 2013

SYNERGY • 2013

DataFlex Web Application Symposium – Part 4

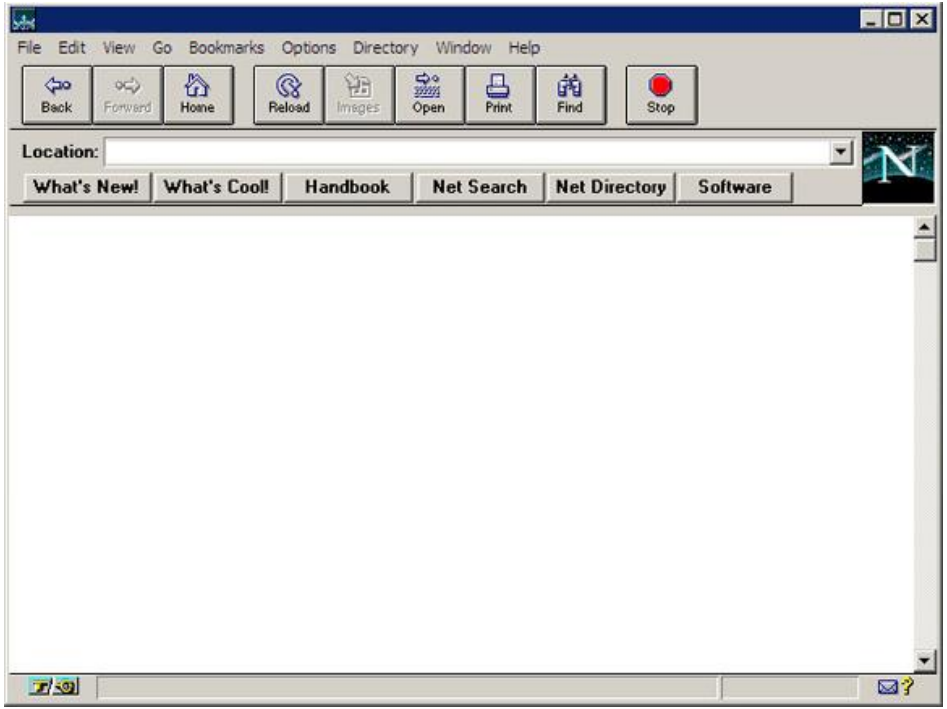
Understanding the Web Framework

Harm Wibier
Development Team
www.dataaccess.com

1993

MOSAIC 0.1

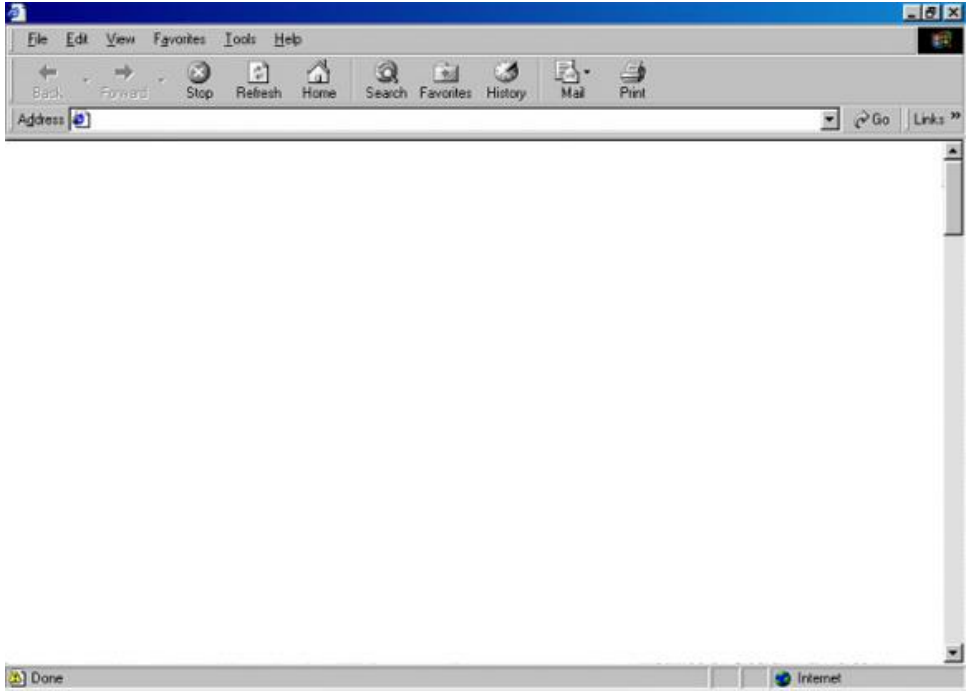
- HTTP
- Hyperlinks
- HTML
- Images



1996

Netscape 2.0

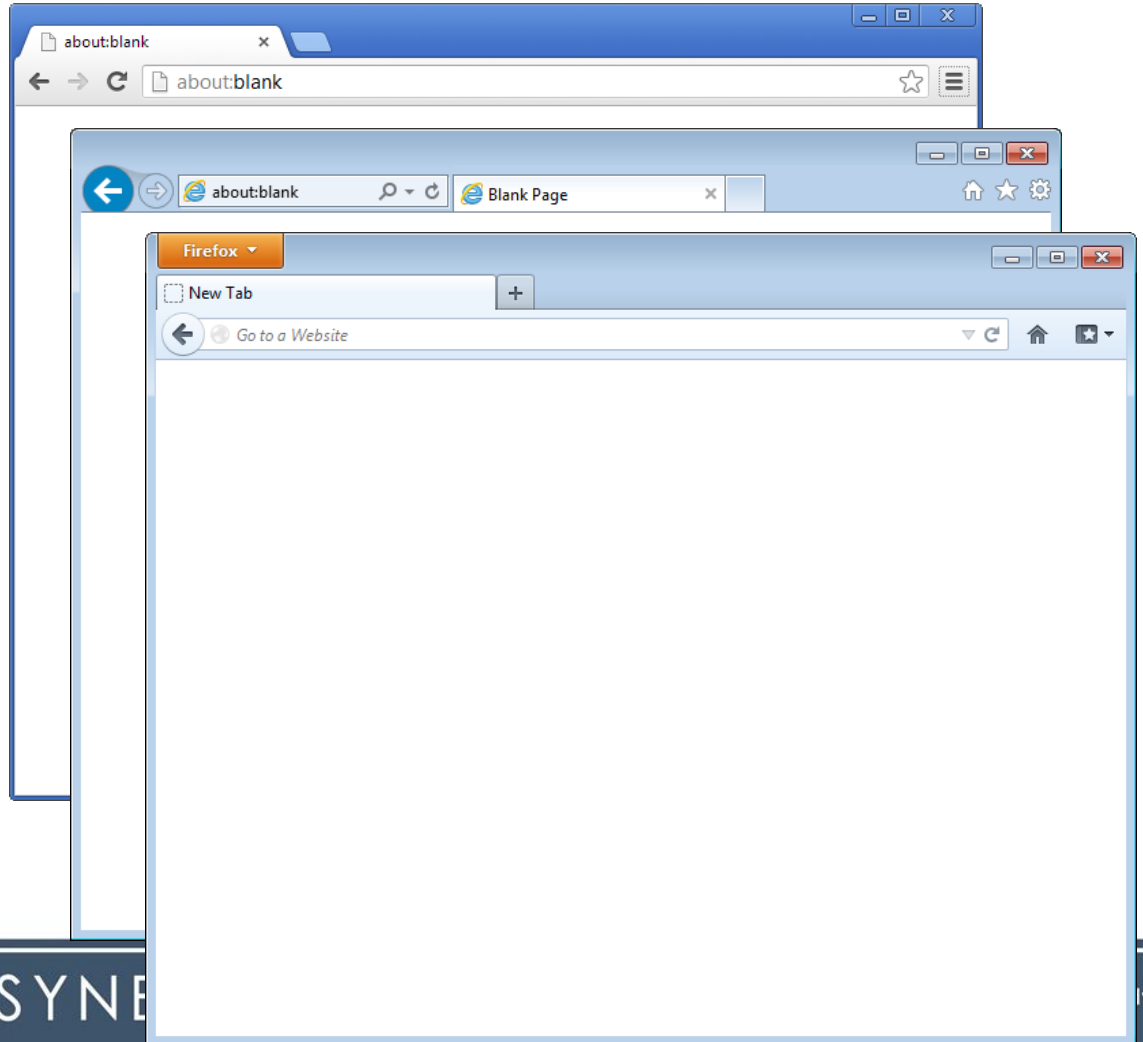
- Forms
- Cookies
- JavaScript



1999

Internet Explorer 5

- CSS
- AJAX (XmlHttpRequest)



2013

Chrome

FireFox

Internet Explorer 10

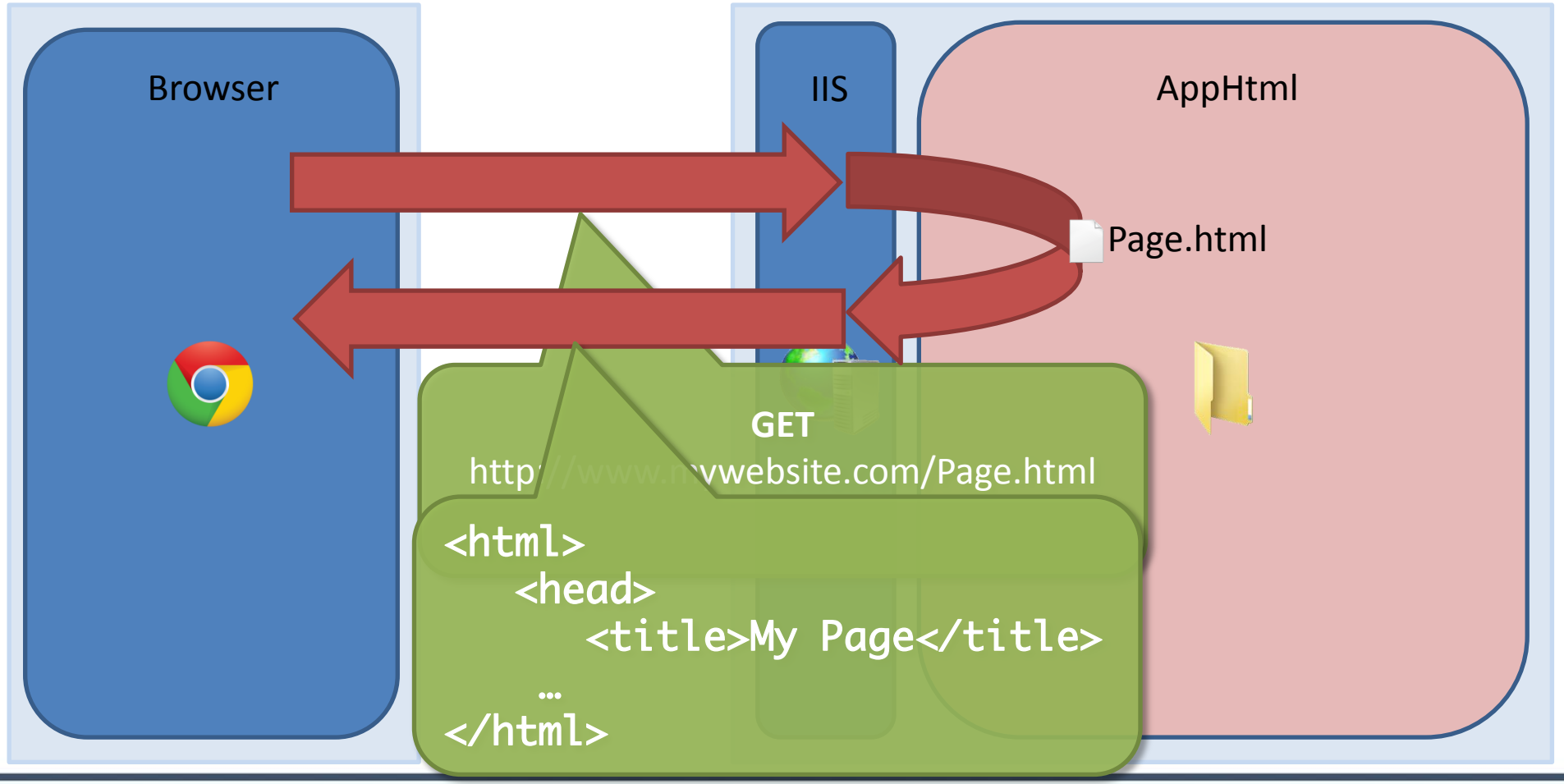
- HTML5
- CSS3

How the web works…

WEB APPLICATIONS

Static pages

- Request is sent to load a page
  - HTML is returned
    - HTTP GET Request
  - Images are loaded
    - HTTP GET Request
- HTML is rendered
  - Includes clickable links to other pages



Classic Web Applications

- Pages become dynamic
  - Server-side scripts
  - Generate HTML
  - Parameters are passed in URL's
    - HTTP GET Request

      customer.asp?customerid=13

  - Data is passed using form posts
    - HTTP POST Request

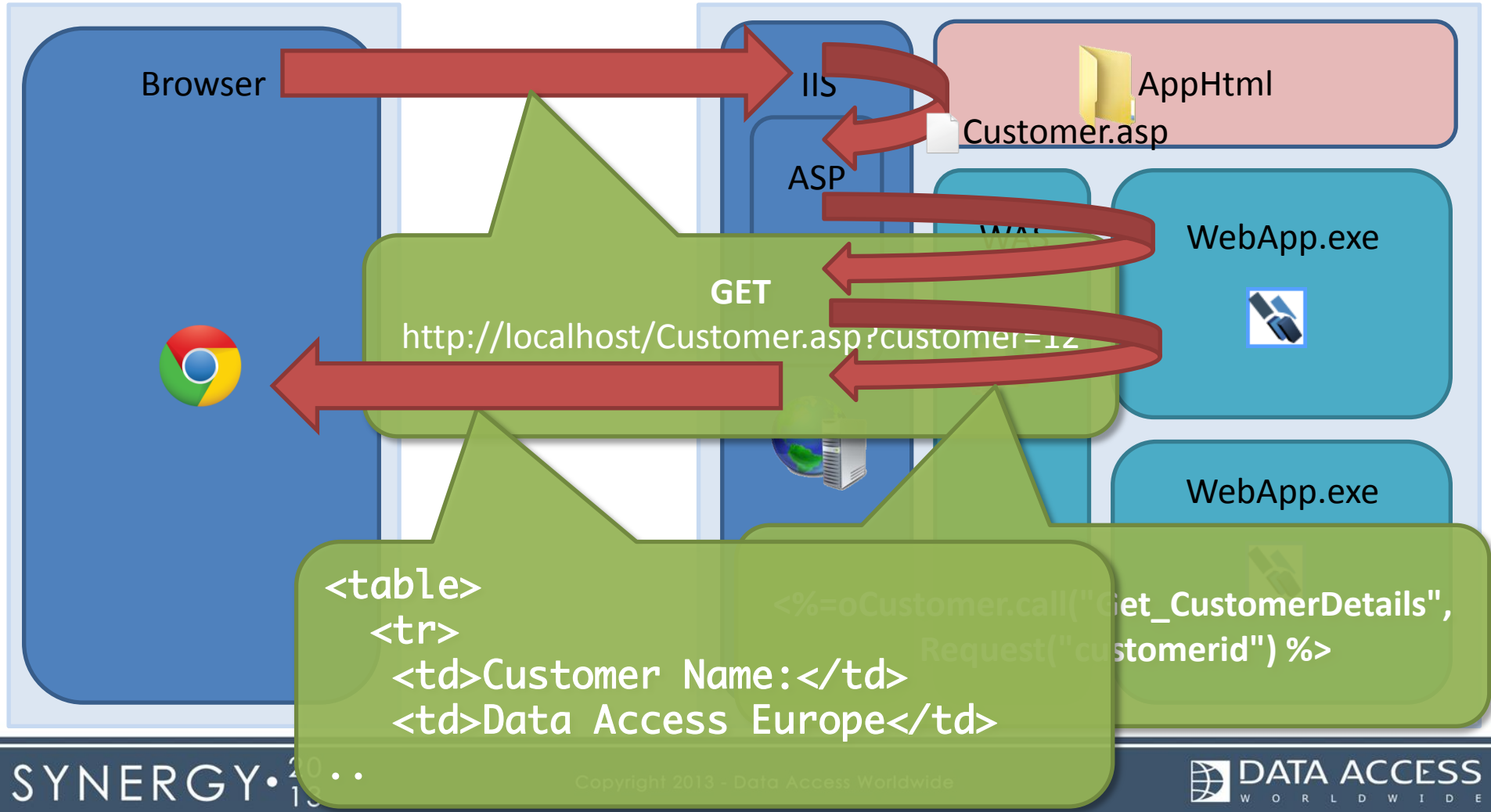
```
<form action="demo_form.asp" method="POST">
```

Web Application Server

- ASP <-> DataFlex
  - Implement Web Business Processes
  - Call published functions from ASP

```
<%=oCustomer.call("Get_CustomerDetails",
   Request("customerid") %>
```

  - GenerateHTML

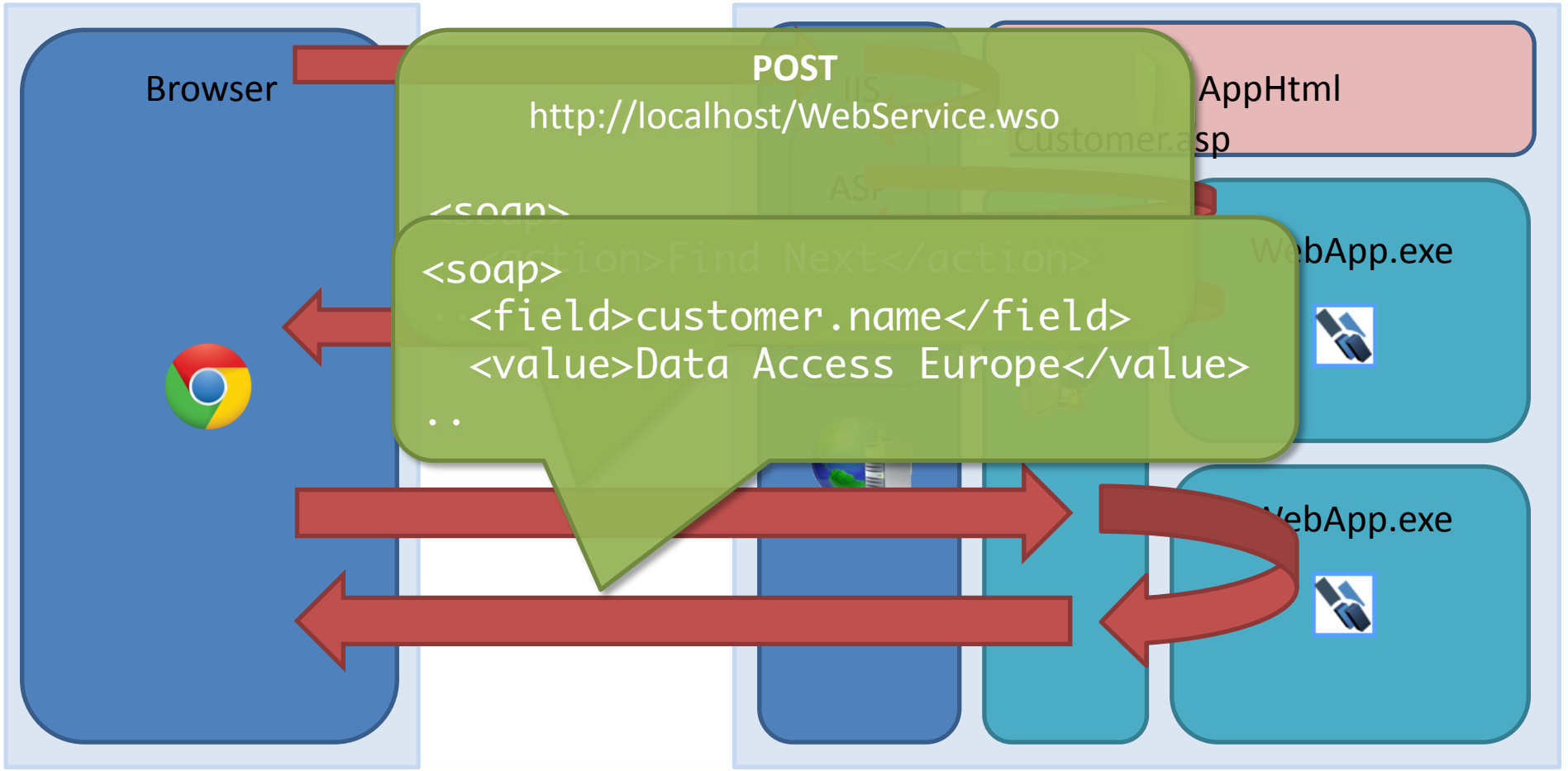


AJAX

- JavaScript makes pages interactive
  - Calls load extra data from server
    - HTTP POST / GET
  - Displayed page is manipulated
    - Document Object Model

AJAX Library

- JavaScript widgets added to HTML
  - Calls DataFlex Web Service
    - HTTP POST
  - Find / Save / Delete Records
  - Emulated DDO's on the client
  - Still uses page reloads
  - Still uses server-side scripts

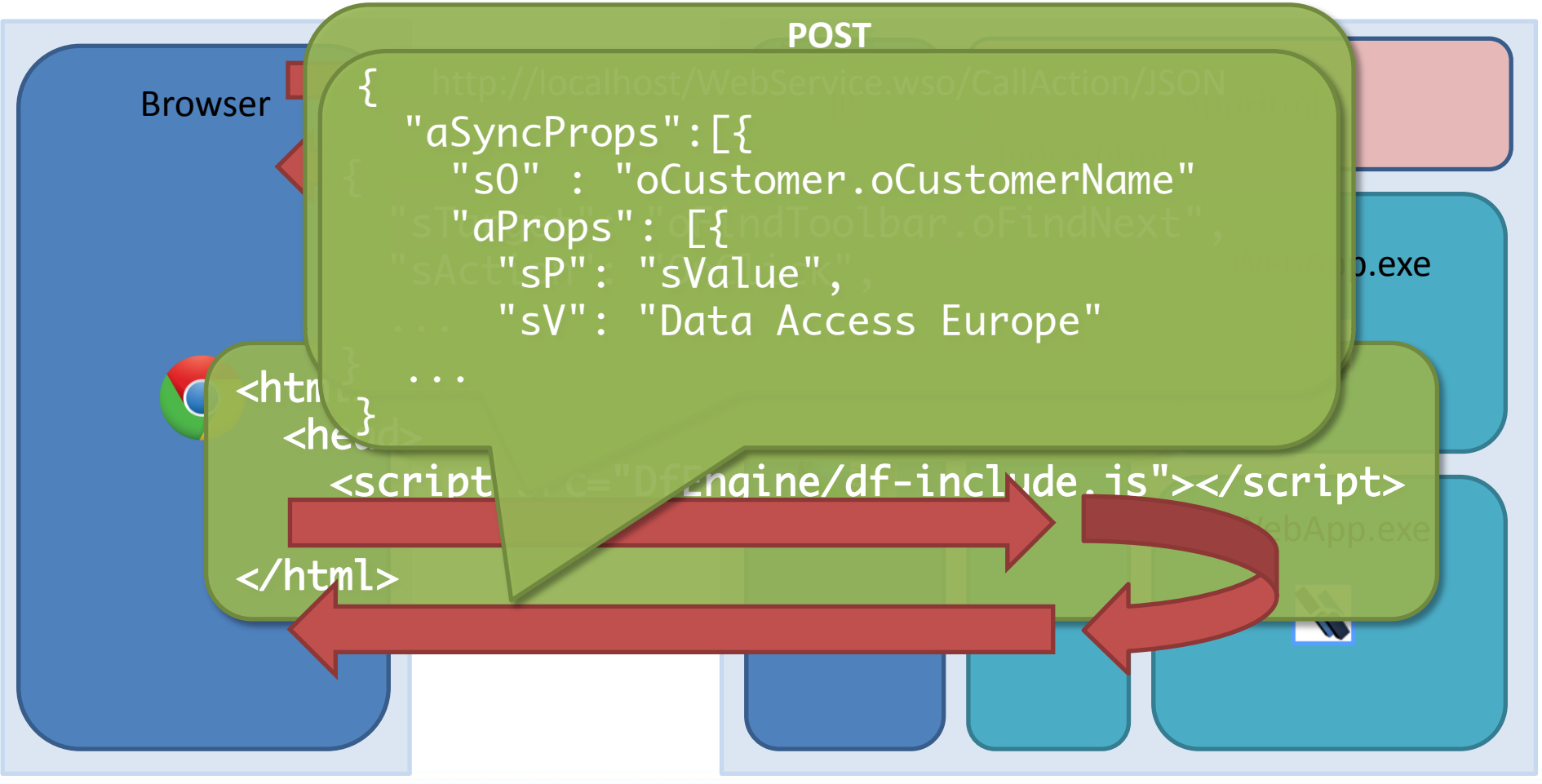


Web 2.0

- More and more client processing
  - Application in a page
    - No more page reload
  - Complete Widget Libraries
  - Client-side storage
  - AJAX, JSON, HTML5, CSS3, …

Web Application Framework

- Application written in Visual DataFlex
  - JavaScript Engine
    - Widget implementation
  - Calls DataFlex Web Service
    - HTTP POST
    - JSON
  - Logic in DataFlex
  - No more ASP



HTTP

- Stateless
  - No connection!
- Calls from the client to the Server
  - GET
    - HTML / CSS / Images
  - POST
    - JSON / XML
    - Web Service calls

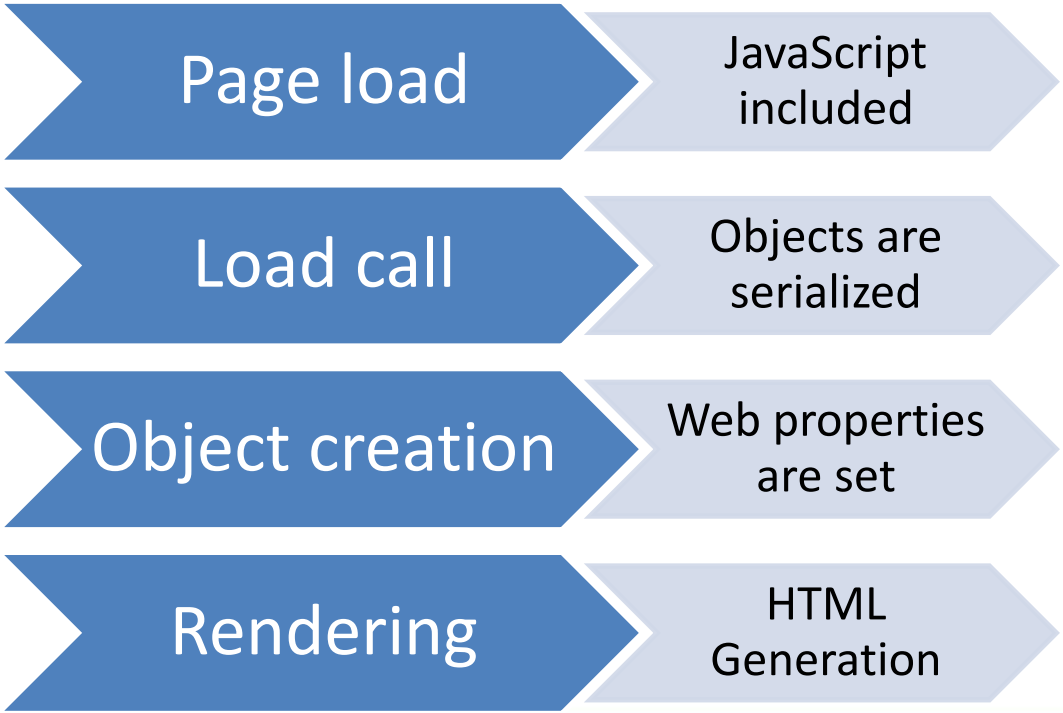
Main concepts of the web framework…

WEB FRAMEWORK CONCEPTS

JavaScript Engine

- Set of classes
  - Classes match DataFlex classes
- Objects are created for web objects
  - Object created for each view, form, menu item, button
- Web properties are implemented
- HTML is generated to render

Initialization



Object serialization

- Object definitions are sent to the client
  - Object name
  - Web properties
  - Class name to be used
  - Nested objects
- Recursive structure
- Implemented in cWebObject

```
{
    "sName": "oCustomerName",
    "sType": "df.WebForm",
    "aProps": [{
        "sN": "pbEnabled",
        "sV": "1"
    },
    {
        "sN": "psLabel",
        "sV": "Name:"
    }],
    "aObjs": []
}
```

Persistency

- No persistency on the server
  - Process pooling
- Calls should be self-explanatory

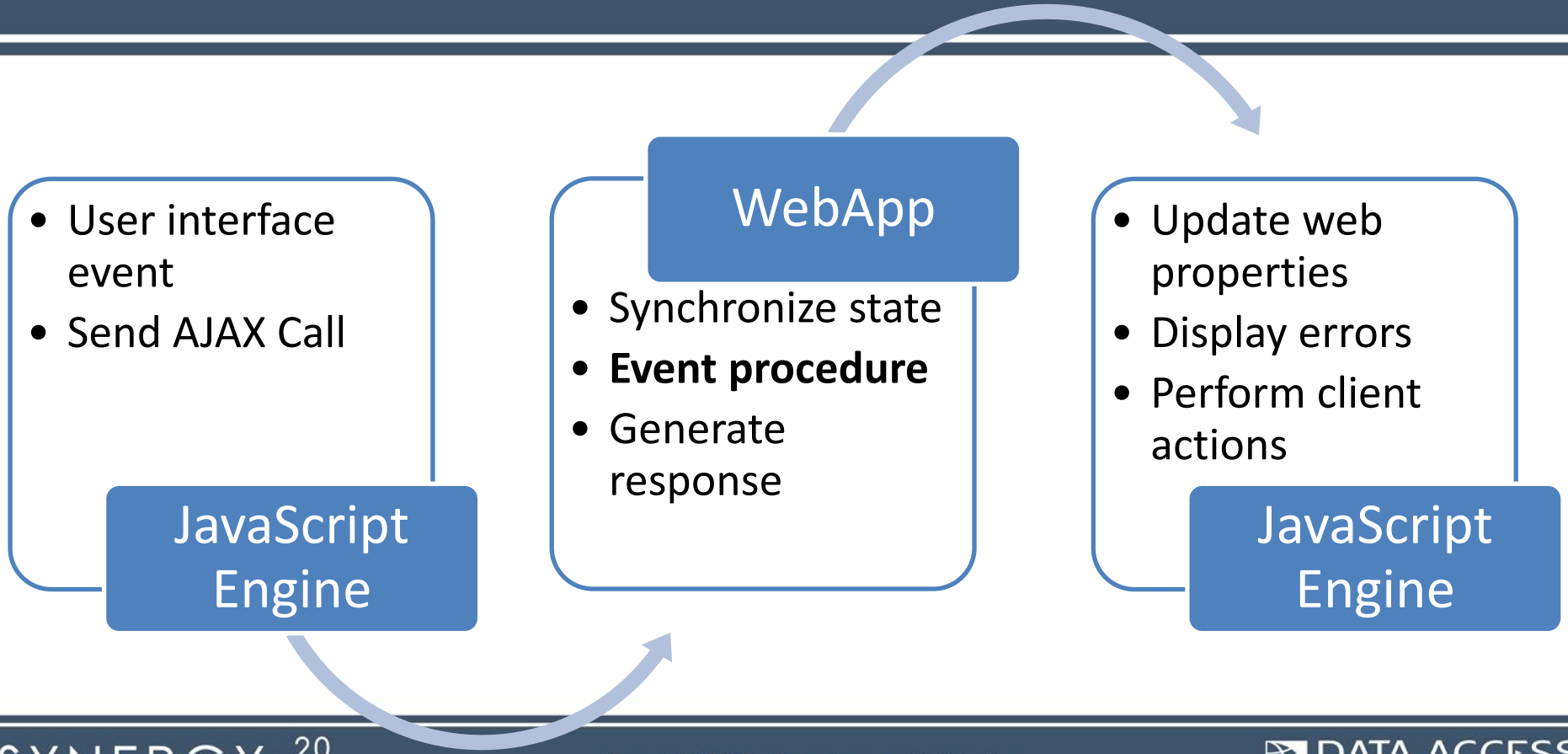
Applications State

- Maintained on the client
  - Web property values
    - List of changed properties
  - DDO Snapshot
    - List of current rowid's per view
- Sent with each call

Events

- Sent to the server
  - Current application state
    - List of changed web properties
    - List of rowid's
  - Server action(s)
- Processed on the server
- Returns
  - Changes on application state
  - Client actions

Server call



John Tuohy will continue from a server-side perspective…

THE DATAFLEX PERSPECTIVE