

XAP:

eXtensible Ajax Platform

Hermod Opstvedt

Chief Architect

DnB NOR ITUD





XAP eXtensible Ajax Platform

- It's an Ajax jungle out there:
 - XAML
 - Dojo
 - Kabuki
 - Rico
 - Direct Web Remoting (DWR)
 - Flex
 - Laszlo
 - + about 30 other more or less ajax enabled frameworks.



XAP eXtensible Ajax Platform

- What is XAP?
 - Originally developed by Nexaweb.
 - XML-based declarative framework for building, deploying and maintaining rich, interactive Ajax powered web applications.
 - Provides an abstraction and normalization layer between the underlying implementation (*dojo, etc.*) and the developer.



XAP eXtensible Ajax Platform

- Described by eXtensible Application Language (XAL) files containing an XML-based markup language.
- XAP maps XAL markup to specific UI widgets.
- No specific server requirements, because XAP code operates only on the client.
 - Need server side XML/JSON for data.
- Runs in any(?) web browser.



XAP eXtensible Ajax Platform

- Leverages existing Ajax projects such as Dojo.
- The widget sets can be extended.
- Has Layout managers.
- Aims to reduce the need for scripting and help solve the development and maintenance challenges associated with large scale JavaScript programming.



XAP eXtensible Ajax Platform

- Because it encapsulates the implementation, you benefit from it
 - Easy to script
 - Simpler to build, once you understand the syntax!
 - There are visual IDE plugins available (1 commercial, none free)
- Has its own UI DOM



XAP eXtensible Ajax Platform

- XAP DOM
 - Separate from the HTML DOM.
 - Maintains the UI state.
 - Provides APIs for modification and structure listeners.



XAP eXtensible Ajax Platform

- XAP engine
 - Contains an XML parser.
 - Modifications modules (XModify).
 - System service containers.



XAP eXtensible Ajax Platform

- Event Handlers
 - Managed Client Objects (MCO's)
 - JavaScript
 - Macros
 - XAL files
 - JavaScript on the client.



XAP eXtensible Ajax Platform

- Request/response handlers on the server.
 - Java
 - Ruby
 - PHP
 - *etc.*



XAP eXtensible Ajax Platform

- Managed Client Objects.
 - Client side managed code objects written in JavaScript that may contain business logic and can be called via XML handlers.
 - Very similar to Java objects



XAP eXtensible Ajax Platform

- MCO sample
 - Import declaration

```
<mco xmlns="http://openxal.org/core/mco" id="login"  
class="Login" src="login.js"/>
```

- A widget:

```
<xal:button width="200px" height="25px"  
text="Login" onCommand="mco:login.doLogin()" />
```

- The MCO (from login.js)

```
Login = function () {};  
  
Login.prototype.doLogin = function () {  
  
    alert('Someone pushed me');  
}
```



XAP eXtensible Ajax Platform

- Macros

- A segment of stored xml that can be run onCommand from the client as if it came from the server.
- XAP Macros should support arguments *via* something like Java MessageFormat but these don't yet.
- Macros can be executed declaratively in component event handlers.



XAP eXtensible Ajax Platform

- Macro sample

```
<xap.application.Application>.getSession().getMacroContainer().getMacro("aMacro").execute();
```

- Direct:

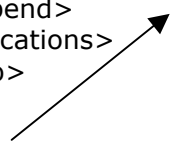
```
<button onCommand="macro:aMacro.execute()"/>
```

- Inline:

- Sample

```
<macro:macro id="aMacro" xmlns:macro="http://openxal.org/core/macro">  
  <xm:modifications>  
    <xm:append select="id('aPanel')">  
      <xal:button text="{0}" y="60%"/>  
    </xm:append>  
  </xm:modifications>  
</macro:macro>
```

Note the param!





XAP eXtensible Ajax Platform

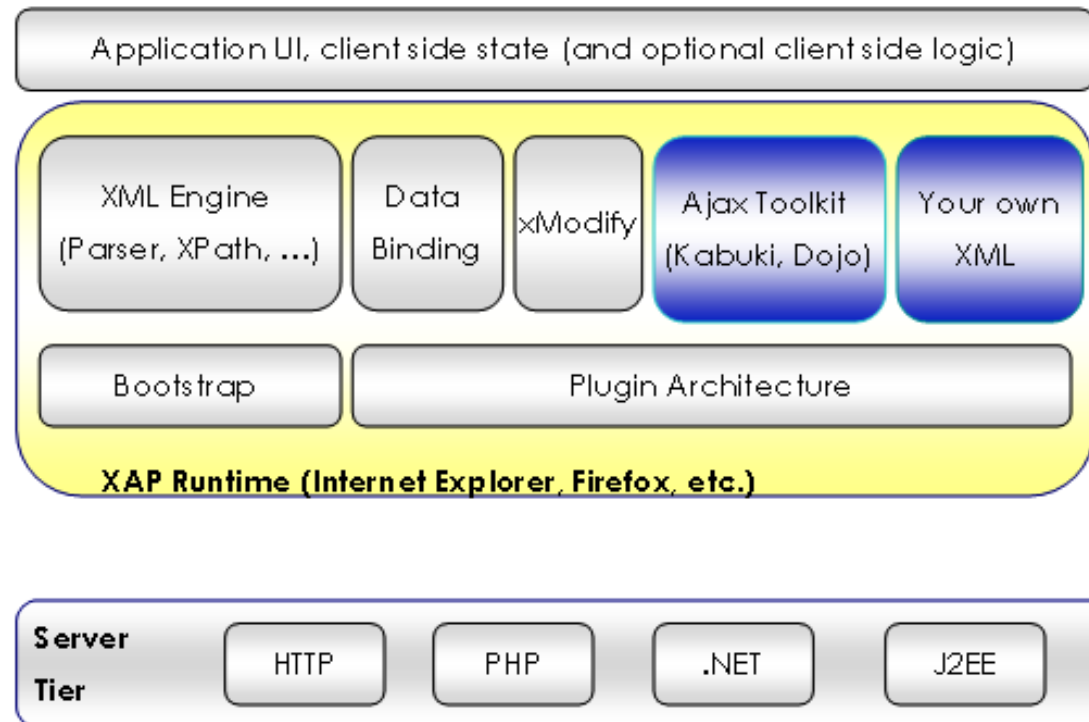
- JavaScript

- Inline javascript in widget onCommand's:

```
<button width="200px" height="25px" text="Push me"  
onCommand="javascript:alert(' I was pushed ');)" />
```

XAP eXtensible Ajax Platform

- The architecture:





XAP eXtensible Ajax Platform

- xModify

- Is used to create or update XML, XAL, and HTML elements declaratively. This means that developers can utilize server-side frameworks to create xModify documents that an xModify Processor will use to update the HTML or XAL document.



XAP eXtensible Ajax Platform

- Several xModify processors (jQuery, MooTools and XAP) to handle the execution of xModify operations.
- The XAP core functionality includes the ability to execute xModify scripts against either the HTML or UI DOMs.



XAP eXtensible Ajax Platform

- Use xModify to change the UI document that contains the visual components of a XAP application. XAP stores all the UI components in a "shadow" document that developers can interact with via xModify or JavaScript APIs.
- You can serve XModify from the server!



XAP eXtensible Ajax Platform

- xModify Operations

- Append:

- Appends its arguments at the end of the children list of each of the results returned by the XPath query.
 - Specified as children of the command element.



XAP eXtensible Ajax Platform

➤ Clone:

- Clones each of the results returned by the XPath query and returns the set of cloned elements.

➤ Create-document

- Creates documents and register with the DocumentRegistry on the client side.
 - ✓ This element supports no attributes and must be the first element under the <xm:modifications> element



XAP eXtensible Ajax Platform

➤ Insert-after:

- Inserts its arguments after each of the results returned by the XPath query of the command.
 - ✓ Specified as children of the command element.

➤ Insert-at:

- Inserts its arguments at the specified index in the children list of each of the results returned by the XPath query.
 - ✓ Specified as children of the command element.



XAP eXtensible Ajax Platform

➤ Insert-before:

- Inserts its arguments before each of the results returned by the XPath query of the command.
 - ✓ Specified as children of the command element.

➤ Remove-attribute:

- Removes the attributes returned by the XPath query of the command.



XAP eXtensible Ajax Platform

➤ Remove-element:

- Removes the elements returned by the XPath query of the command.

➤ Replace:

- Replaces each of the results returned by the XPath query with the command arguments
 - ✓ Specified as children of the command element.



XAP eXtensible Ajax Platform

➤ Replace-children:

- Removes the elements children for each element returned by the XPath query.
 - ✓ The command arguments are inserted in their place.

➤ Set-attribute:

- Sets the specified attributes in all of the elements returned by the XPath query.
 - ✓ All attributes to be set need to be specified as an attribute directive element.



XAP eXtensible Ajax Platform

- xModify operations that apply only to HTML
 - Set-innerhtml:
 - Takes the contents of the tag and uses it as the value of the `htmlElement.innerHTML`.
 - Set-style:
 - Sets the specified attributes in all of the elements returned by the XPath query.
 - ✓ All attributes to be set need to be specified as an style directive element.



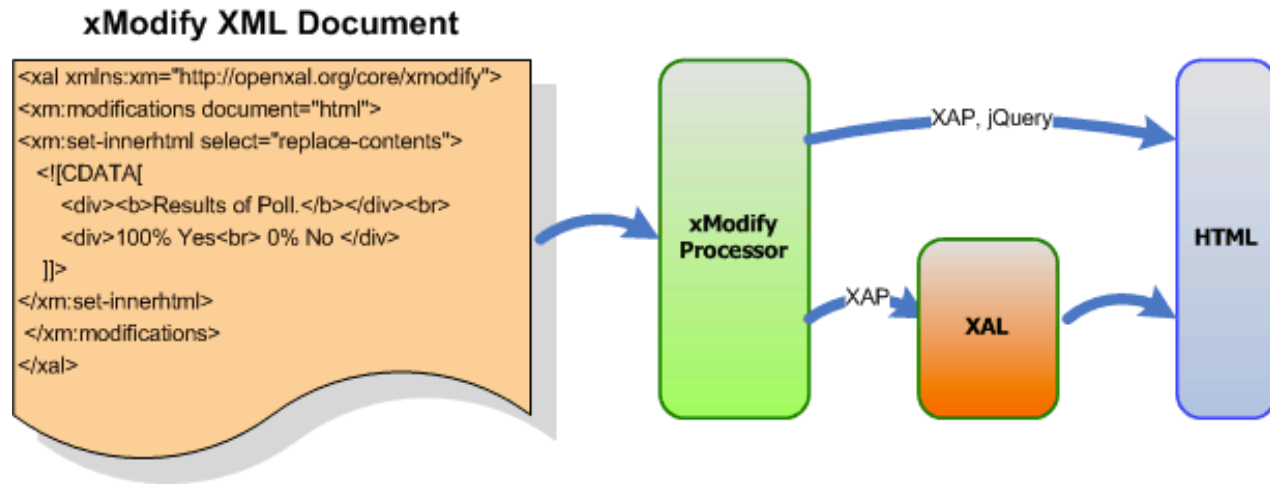
XAP eXtensible Ajax Platform

➤ Embed-xal:

- Used to change the elements selected *via* the "select" attribute into a XAL host.
- Typically used when you are using HTML as the layout mechanism for your application and you want to place the XAL widgets within a specific location.

XAP eXtensible Ajax Platform

- Architecture:





XAP eXtensible Ajax Platform

- Databinding
 - Datasources
 - XML, Document or JSON
 - Binding types:
 - ✓ one-way
 - ✓ one-time
 - ✓ two-way



XAP eXtensible Ajax Platform

- RequestService

- retrieve

- Retrieves the XMLHttpRequest object from the specified URL.

- retrieveAndProcess

- Retrieves a server-side page and processes it assuming it is proper XML, executing any namespaced commands like Xmodify.



XAP eXtensible Ajax Platform

- retrieveAndProcessAsynchronously
 - Asynchronously retrieves the XMLHttpRequest object from the specified URL.
- urlToPath
 - Given a url like: /a/b?c=d Turn it into a path ending in slash like /a/.



XAP eXtensible Ajax Platform

- Formatting
 - ✓ MessageFormatter
 - ✓ DecimalFormatter
 - ✓ DateFormatter
 - ✓ CurrencyFormatter
- Formatters can be chained



XAP eXtensible Ajax Platform

- Widgets

- Layout managers such as:

- Freepanel

- Containers such as:

- Window
 - Panel
 - *etc.*



XAP eXtensible Ajax Platform

- Standard widgets such as:
 - Table
 - Tree
 - Button
 - *etc.*



XAP eXtensible Ajax Platform

- Current widget list
 - ❖ button
 - ❖ label
 - ❖ link
 - ❖ checkBox
 - ❖ radioButton
 - ❖ comboBox
 - ❖ editableComboBox
 - ❖ listBox
 - ❖ textField
 - ❖ textArea
 - ❖ passwordField



XAP eXtensible Ajax Platform

- Widgets

- ❖ textView
- ❖ tabPage
- ❖ horizontalSplitPane
- ❖ verticalSplitPane
- ❖ freePane
- ❖ verticalBoxPane
- ❖ horizontalBoxPane
- ❖ borderPane
- ❖ table
- ❖ row



XAP eXtensible Ajax Platform

- Widgets

- ❖ cell
- ❖ tree
- ❖ treeTable
- ❖ menu
- ❖ popupMenu tooltip
- ❖ window
- ❖ horizontalFlowPane
- ❖ scrollPane
- ❖ desktopPane
- ❖ verticalToolBar
- ❖ horizontalToolBar



XAP eXtensible Ajax Platform

- Extensibility

- 1. Define your tags.
- 2. Create the bridge.
- XAP provides you with
 - Common event model.
 - Component interoperability.
 - An abstracted UI component.
 - Data binding.

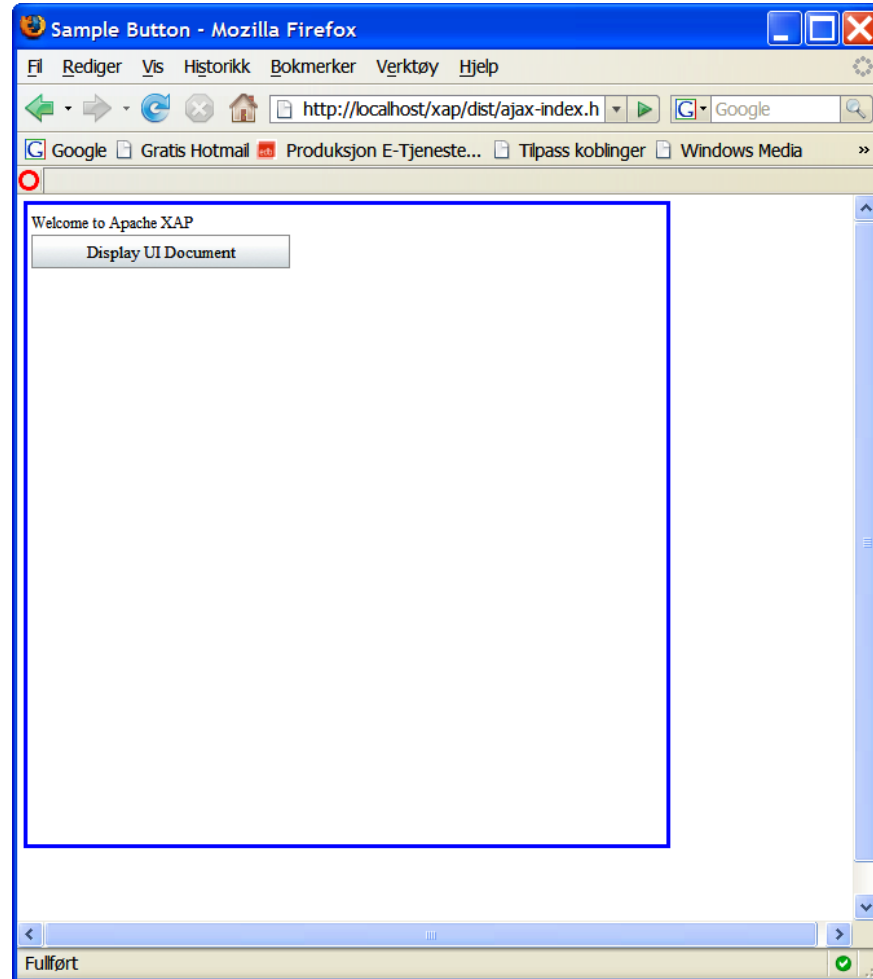


XAP eXtensible Ajax Platform

- Installation

- Download jar file.
- Unzip it to the document root of your web server
 - Ex. Apache httpserver: htdocs/xap
- Verify by opening /dist/ajax-index.html located under the directory you unpacked to (ex xap)

XAP eXtensible Ajax Platform





XAP eXtensible Ajax Platform

- How to use it – First some simple samples

➤ The infamous “Hello World”



XAP eXtensible Ajax Platform

- First we need a template for our samples:
 - Start by adding the following to the HEAD section:

```
<script type="text/javascript">  
  djConfig = {  
    parseWidgets: false  
  };  
</script>
```

```
<script language="JavaScript" type="text/javascript" src="../../xapcore.js"></script>
```



XAP eXtensible Ajax Platform

➤ Looking at the declaration:

```
<script type="text/javascript">  
  djConfig = {  
    parseWidgets: false  
  };  
</script>
```

➤ This initializes XAP and dojo.



XAP eXtensible Ajax Platform

```
<script language="JavaScript" type="text/javascript" src="xap/lib/xapcore.js"></script>
```

➤ Loads the core XAP library

```
<script language="JavaScript" type="text/javascript">  
  function onLoad(){  
    Xap.createEmbeddedApplications();  
  }  
</script>
```

➤ Initializes XAP

- Another alternative is to use the onLoad attribute of the body tag.

✓ <body
 onload="Xap.createEmbeddedApplications();">



XAP eXtensible Ajax Platform

➤ In the BODY section add your stuff:

```
<body onload="onLoad();">  
  <div applicationName="HelloWorld" startPage="helloworld.xal" configFile="xap/config/XapConfig.js"></div>  
</body>
```



XAP eXtensible Ajax Platform

➤ The complete html:

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>CSS2007 Hello World</title>
    <script type="text/javascript">
        djConfig = {
            parseWidgets: false
        };
    </script>
    <script language="JavaScript" type="text/javascript" src="../../xapcore.js"></script>
</head>
<body onload="Xap.createEmbeddedApplications();">
    <div applicationName="Hello World" startPage="./HelloWorld.xal" configFile="../XapConfig.js"></div>
</body>
</html>
```



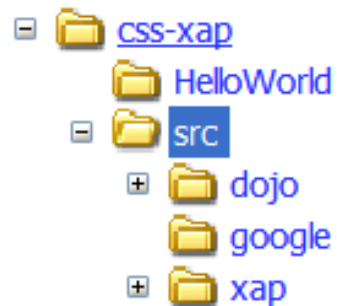
XAP eXtensible Ajax Platform

➤ The xal document (helloworld.xal):

```
<xal xmlns="http://openxal.org/ui">  
  <xm:modifications xmlns:xm="http://openxal.org/core/xmodify">  
    <xm:append select="/ui">  
      <label text="Hello World (CSS 2007)" width="100%"/>  
    </xm:append>  
  </xm:modifications>  
</xal>
```

XAP eXtensible Ajax Platform

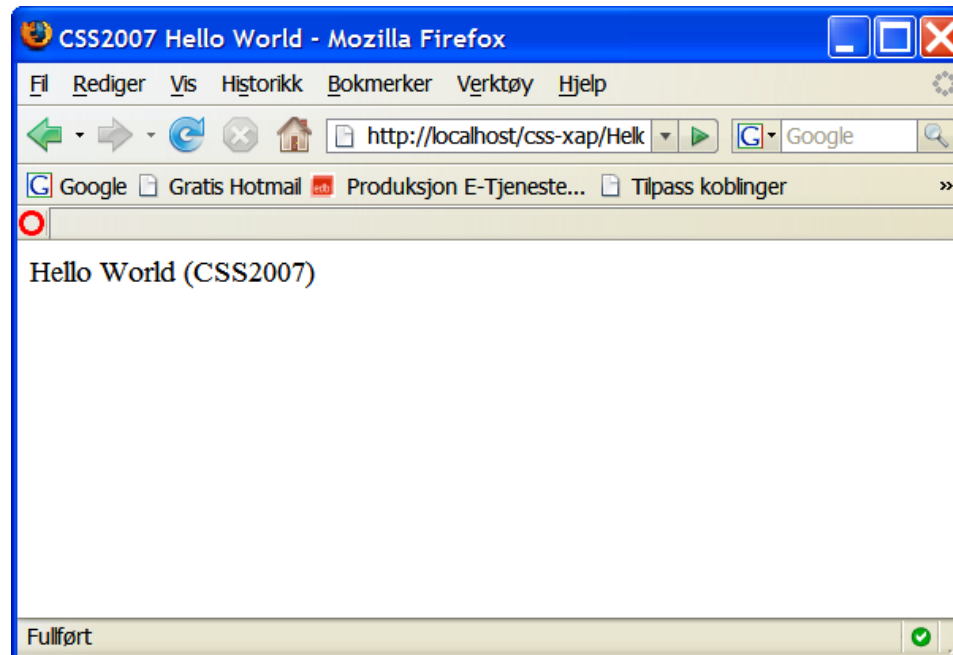
- Then we need to deploy this to our http server along the necessary XAP files.



- Remember to add any changes to xapConfig.js

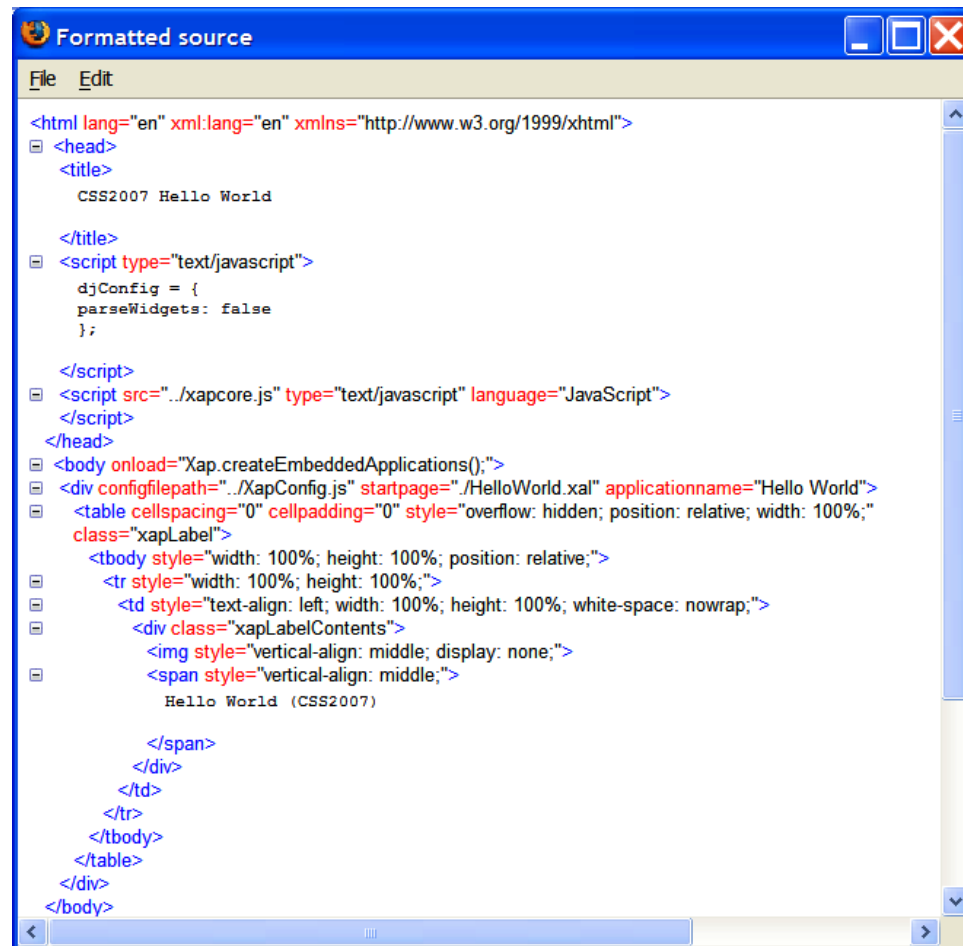


XAP eXtensible Ajax Platform



XAP eXtensible Ajax Platform

- The rendered html:



```
<html lang="en" xml:lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>
  CSS2007 Hello World
</title>
<script type="text/javascript">
  djConfig = {
    parseWidgets: false
  };
</script>
<script src="../../xapcore.js" type="text/javascript" language="JavaScript">
</script>
</head>
<body onload="Xap.createEmbeddedApplications();">
<div configfilepath="../../XapConfig.js" startpage="/HelloWorld.xal" applicationname="Hello World">
<table cellspacing="0" cellpadding="0" style="overflow: hidden; position: relative; width: 100%;"
  class="xapLabel">
  <tbody style="width: 100%; height: 100%; position: relative;">
  <tr style="width: 100%; height: 100%;">
  <td style="text-align: left; width: 100%; height: 100%; white-space: nowrap;">
  <div class="xapLabelContents">
  <img style="vertical-align: middle; display: none;">
  <span style="vertical-align: middle;">
    Hello World (CSS2007)
  </span>
  </div>
  </td>
  </tr>
  </tbody>
  </table>
</div>
</body>
</html>
```





XAP eXtensible Ajax Platform

- Let's add some more stuff (input fields, *etc.*)
- Create a new html file sample2.html based on our standard template.
- Change the body to use sample2.xal



XAP eXtensible Ajax Platform

- Sample2.xal

```
<xal xmlns="http://openxal.org/ui" xmlns:xal="http://openxal.org/ui">
  <xm:modifications xmlns:xm="http://openxal.org/core/xmodify">
    <xm:append select="/ui">
      <freePane width="500px" height="300px"
        backgroundColor="#F0F0F0" id="testComponentFreePane"
        borderStyle="solid" borderWidth="2px" borderColor="#000">
        <xal:verticalBoxPane>
          <label text="Username" />
          <textField id="username" />
          <label text="Password" />
          <textField id="password" />
        </xal:verticalBoxPane>
      </freePane>
    </xm:append>
  </xm:modifications>
</xal>
```





XAP eXtensible Ajax Platform

- Some more samples....
 - More complex ui's ...
 - Server side integration...



XAP eXtensible Ajax Platform

- Why you may want to use it:
 - For security reasons – Ajax is not secure
 - You like JavaScript.
 - You are familiar with XPath for server side integration
 - You don't need WISIWYG (read not paying for it)
 - You like working with XML



XAP eXtensible Ajax Platform

- Why you may not want to use it
 - Don't want to learn XPath.
 - Steep learning curve.
 - No open-source IDE integration
 - NexaWeb has a commercial IDE plugin product



XAP eXtensible Ajax Platform

- References:

- <http://incubator.apache.org/xap/index.html>

- <http://www.nexaweb.com/home/us/index.html>

- Cool online demos:

- <http://www.rockstarapps.com/>



XAP eXtensible Ajax Platform

