

Building Ajax Applications with GT.M and EWD

Rob Tweed

M/Gateway Developments Ltd



Is GT.M old-fashioned?

- “I am a physician working in the VA system. I also program in .net. Certainly the VAs informatics back bone “VISTA” is impressive...for the 1980s. “

Is GT.M old-fashioned?

- In fact that's just a reflection of when VistA was developed
- The fact is that GT.M is not only the right kind of database for today's applications..
- It also supports all the very latest Ajax techniques and frameworks.
- One of the best platforms available!
- Eg...

What is Ajax?

- A term coined by Jesse James Garrett in 2005 to describe a range of techniques and technologies that were starting to appear widely:
 - *<http://adaptivepath.com/publications/essays/archives/000385.php>*
- Ajax = Asynchronous Javascript and XML
- Actually XML not often used at all!
 - Broad and vague term that describes a style of application more than a set of technologies

What distinguishes an Ajax application?

- Not page-oriented, breaks away from:
 - Fill out a form
 - Click a button
 - Page refreshed with another
- Ajax is event-oriented - content dynamically changed within a single page
 - Some Ajax applications are just a single page of HTML that the user interacts with for hours!
 - Looks and behaves much more like a typical client/server VB or Java application

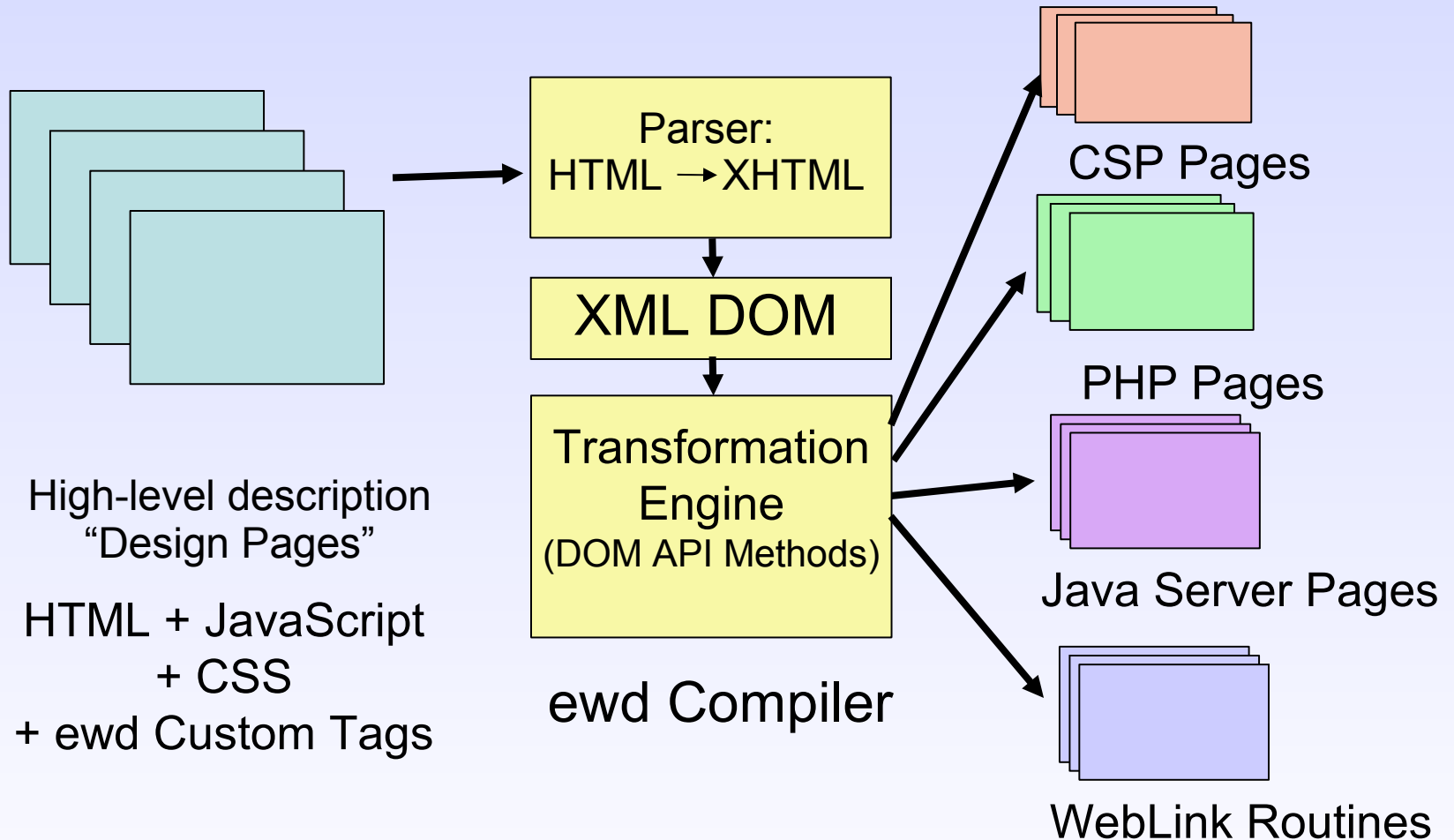
What is EWD?

- Enterprise Web Developer
- A technology-agnostic framework for developing web and Ajax applications
- Design-focused
- Reduces programming to the minimum
- Automates as much as possible
- XML-based descriptions
 - What, not how

EWD technology independence

- “front end”
 - PHP, JSP, ASP.Net, CSP, WebLink
 - Python, Ruby to come
- “back-end”
 - GT.M, Caché, MySQL

EWD's Compiler



EWD and GT.M

- EWD is available for GT.M
- Free (but not currently FOSS)
- Available in various formats:
 - Suite of Mumps routines
 - Install kit for Asus Eee
 - Pre-packaged Virtual Appliance
 - VMWare
 - Ubuntu Linux JEOS
 - GT.M
 - EWD
 - MGWSI gateway
 - PHP
 - Apache

Who uses EWD?

- Quest Diagnostics
 - Care360 test requests/reporting
 - 0.5 million tests per night
 - 30,000 concurrent users
- Goodfellow UK online ordering
- Dept of Veterans Affairs, audiology
- Etc...

Let's take a look

- EWD Virtual Appliance
- A typical Ajax application
 - ewdMgr application
 - Pre-built EWD/Ajax application
 - Managing your EWD environment

Ajax Key Technologies

- Javascript and Event Handlers
- Cascading Style Sheets (CSS)
- XMLHttpRequest Object
 - “back-channel” built into the browser that allows a Javascript function to communicate externally via HTTP
 - Standard HTTP request/response
- Browser’s Document Object Model (DOM)
 - Dynamic HTML
- JSON (JavaScript Object Notation)

Typical Ajax technique

- Catch an in-browser event:
 - Standard Javascript event handler
 - onClick, onChange, onMouseOver
- Send an asynchronous request via XMLHttpRequest object
- Use the response to modify part of the page in the browser by manipulating the DOM
 - Or send and optionally invoke a JavaScript object via JSON

Browser's DOM

- When a page of HTML is received by a browser, it is parsed into a DOM
- API methods to manipulate the DOM are implemented in Javascript
 - `document.getElementById("myField").parentNode`
 - `document.getElementById("myField").firstChild`

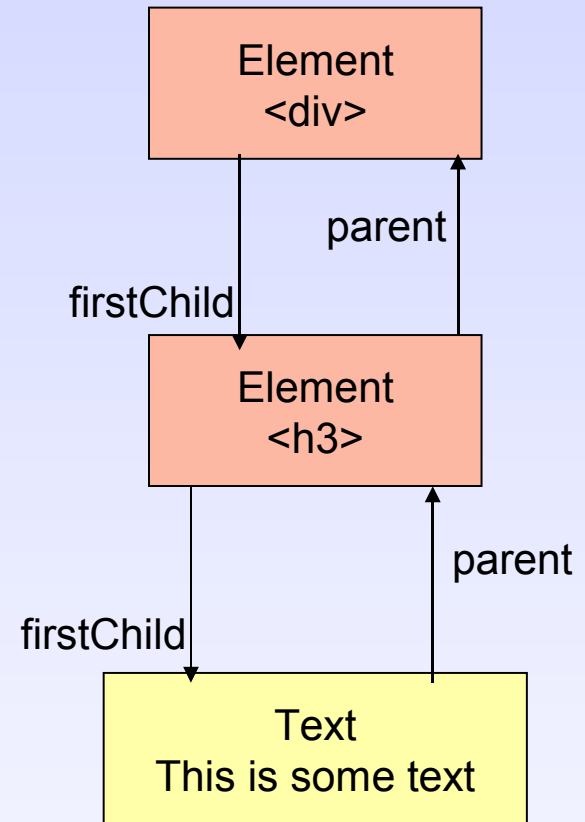
Javascript DOM API

Example:

```
var alertPointer = document.getElementById("ajaxAlert") ;  
var head = document.getElementsByTagName("head").item(0);  
alertPointer = document.createElement("script") ;  
alertPointer.id = "ajaxAlert" ;  
alertPointer.type = "text/javascript" ;  
head.appendChild(alertPointer) ;  
alertPointer.text = "EWD.ajax.errorOn() ;" ;
```

DOM Content Replacement

```
<div id="myId">  
  <h3>  
    This is some text  
  </h3>  
</div>
```



DOM Content Replacement

```
<div id="myId">
```

```
<h3>  
  This is some text  
</h3>
```

```
</div>
```



```
document.getElementById("myId").innerHTML
```

- read/write property

DOM Content Replacement

```
document.getElementById("myId").innerHTML = "<p>Some new text</p>" ;
```

```
<div id="myId">
```

```
<h3>
```

```
  This is some text
```

```
</h3>
```

```
</div>
```

DOM Content Replacement

```
document.getElementById("myId").innerHTML = "<p>Some new text</p>" ;
```

```
<div id="myId">
```

```
</div>
```

DOM Content Replacement

```
document.getElementById("myId").innerHTML = "<p>Some new text</p>" ;
```

```
<div id="myId">
```

```
<p>
```

Some new text

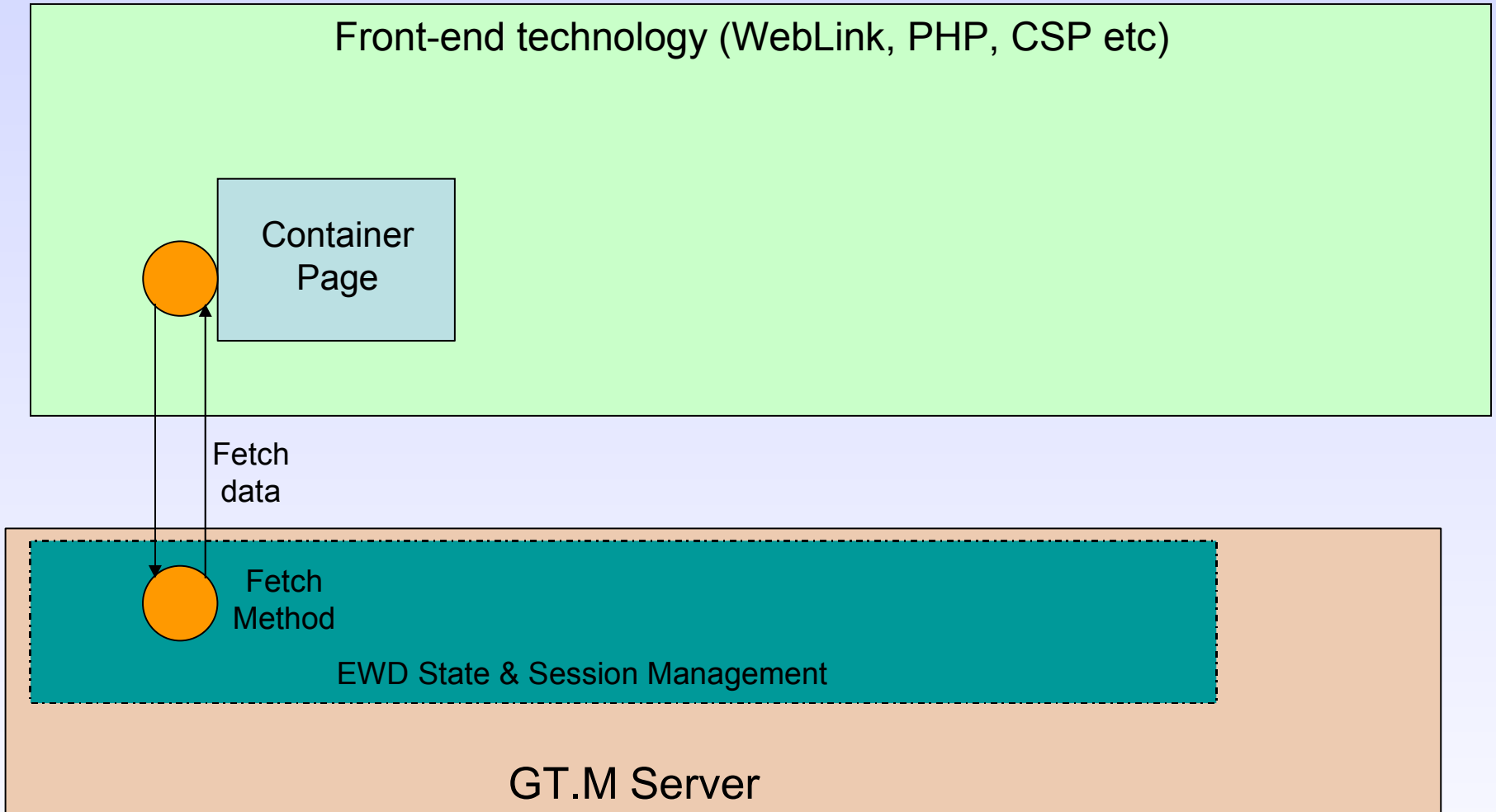
```
</p>
```

```
</div>
```

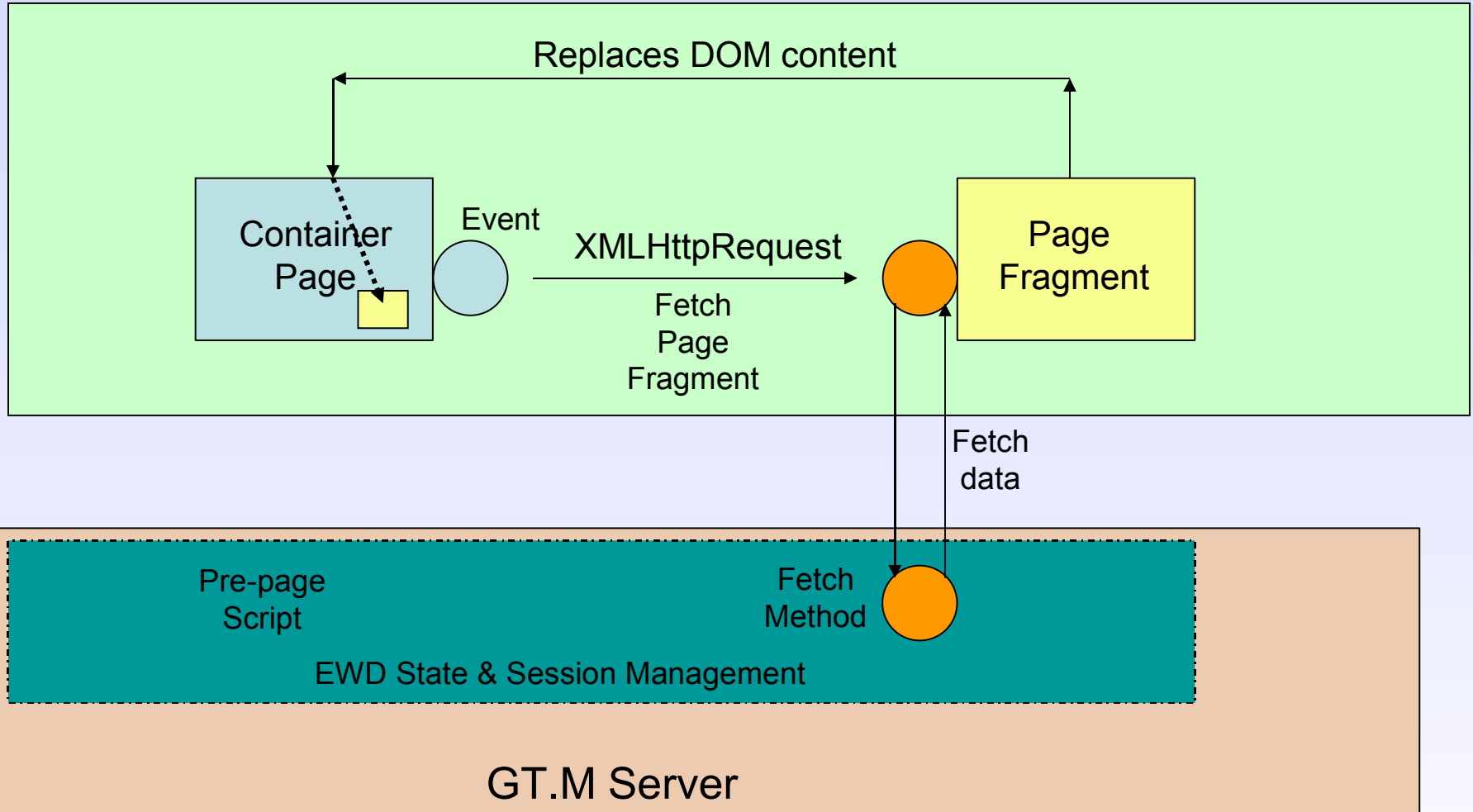
Browser DOM is active

- As soon as you make a modification to the browser's DOM, the browser re-renders the page
- By replacing DOM content with markup delivered as page fragments, we can create an interactive experience for the user
- EWD makes this easy...

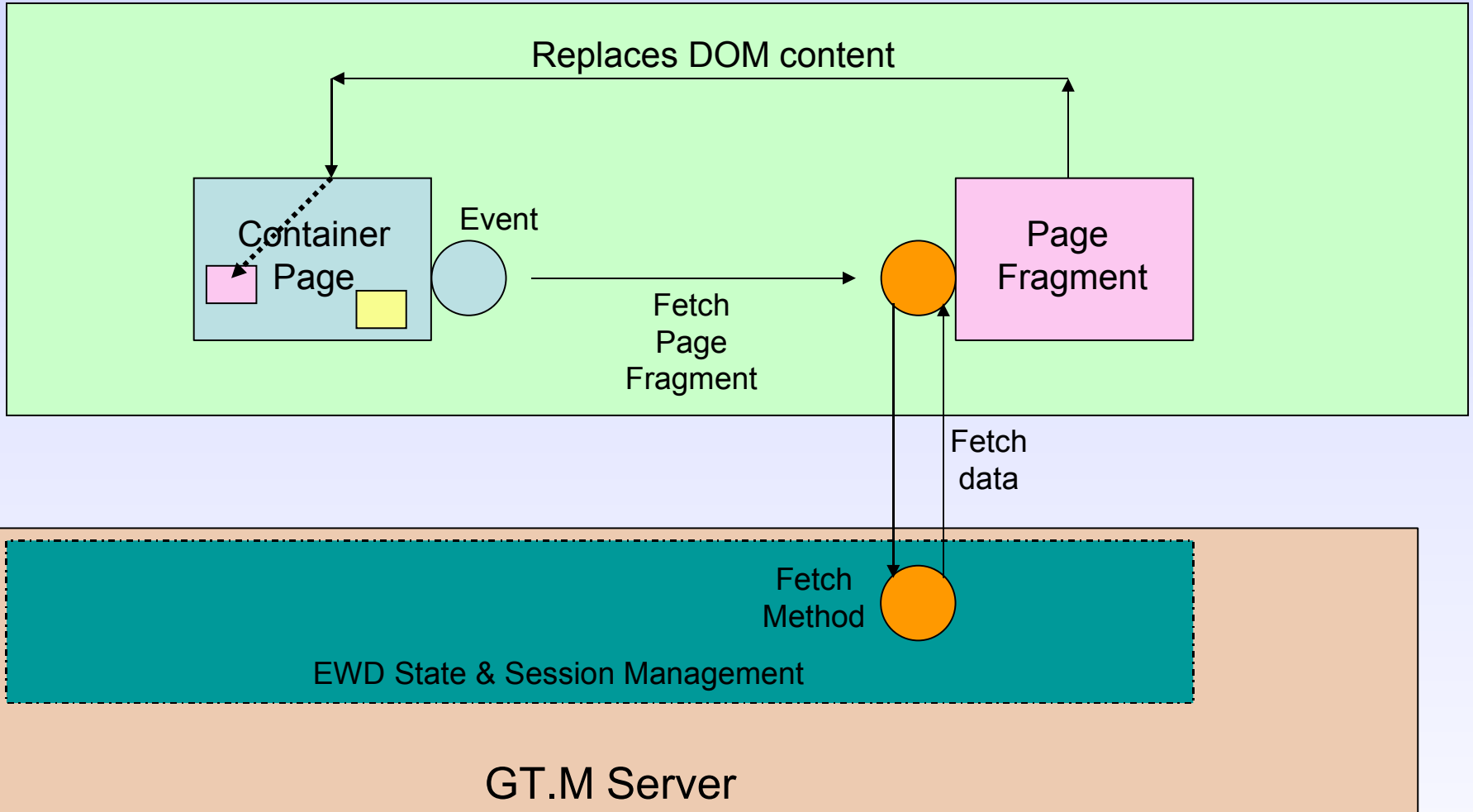
Ajax Event Cycle (EWD)



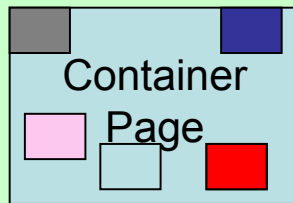
Ajax Event Cycle (EWD)



Ajax Event Cycle (EWD)



Ajax Event Cycle (EWD)



EWD State & Session Management

GT.M Server

Example page

- From ewdMgr
 - mainMenu.ewd
 - Very high-level description of what the page does
 - Written using Custom Tags
 - Compiler.ewd
 - Ewd's form automation
 - Cross-reference
 - Event
 - Fragment name
 - targetId

Developing with EWD

- EWD Virtual Appliance
 - Apache
- WinSCP
- puTTY
- Serenji editor (optional)
- Browser

Developing with EWD

- Create new application directory
 - Under */usr/ewdapps*
 - “application root directory”
- Create a page using text editor
 - Eg within WinSCP or using Serenji
 - File extension *.ewd*
- Compile application
 - Using *ewdMgr*
 - Command prompt

Developing with EWD

```
ewd@ubuntuJEOS: /usr/local/gtm/ewd
ewd@ubuntuJEOS:/$
ewd@ubuntuJEOS:/$
ewd@ubuntuJEOS:/$ cd /usr/local/gtm/ewd
ewd@ubuntuJEOS:/usr/local/gtm/ewd$ $gtm

GTM>d compilePage^%zewdAPI("extjs","formDemo1a",,"php")
/usr/ewdapps/extjs/ewdAjaxError.ewd
/usr/ewdapps/extjs/ewdAjaxErrorRedirect.ewd
/usr/ewdapps/extjs/ewdErrorRedirect.ewd
/usr/ewdapps/extjs/formDemo1a.ewd

GTM>

GTM>

GTM>

GTM>

GTM>

GTM>

GTM>
```

Developing with EWD

- Running in a browser
- <http://192.168.1.10/php/helloWorld/hello1.php>

Application name

Page name
(derived from hello.ewd)

A quick “hello world”

- Let’s build it...

Developing with EWD

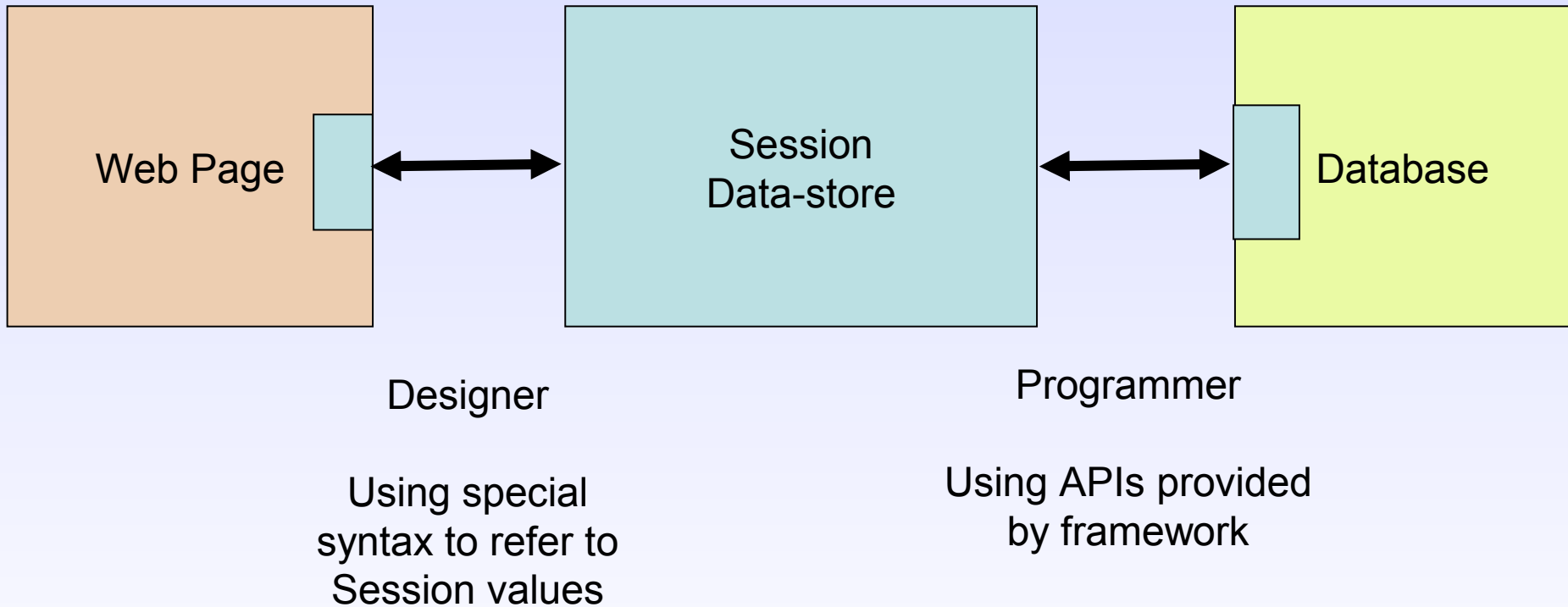
- Fetch methods
 - Create Mumps routine in /usr/local/gtm/ewd
 - Compile using mumps *.m

Interfacing design and programming

- Need a conceptual interface through which the designer and programmer can inter-operate
- This is provided by the “Session” data store

The Session Interface

Provides the interface between the designer and the programmer



No data binding!

- Data binding is a scourge!
 - It begins with a blessing
 - But it ends in a curse
- The Session data-store de-couples the page's data from the physical database

“hello world” Mark 2

- Simple fetch method
- Simple session variable
- Let's build it...

Developing with EWD

- Typically an EWD Ajax application is a single page as far as the browser is concerned
 - Fragments of the DOM are altered
 - Page never leaves the browser

Security Management

- EWD's compiler and run-time automatically tokenises all URLs
 - Randomly-generated character strings
- Controller rejects URLs with invalid tokens
- Tokens provide pointer to the user's session
- Completely automatic

Designer just specifies: ``

EWD creates at runtime, eg:

``



Custom Tags

- EWD's functionality can be extended by defining XML-based custom tags
- You define the XML DOM transformations that the compiler will make to a page when encountering an instance of your custom tag
- Hugely powerful
- Has been used to create very easy-to-use custom tags for ExtJS, Emprise JS Charts, Dojo and YUI
 - Describe what you want, eg a grid
 - Not how, ie the associated complex Javascript
- Further encourages standardisation and discipline because they save time

Multi-lingual Deployment

- EWD's compiler can automatically detect all text in a page
 - Creates and maintains text/phrase database
 - Substitutes call-backs in run-time page
 - Just get text/phrases translated
 - At run-time, call-backs in page retrieve translated text
- Developers continue to design using their native language
 - No impact on development
 - Can be done retrospectively for existing EWD pages
- Built-in contextual review mode

Javascript Ajax Frameworks

- ExtJS
- Complete Javascript framework
- Many sophisticated “widgets”
- EWD supports ExtJS
 - Set of custom tags
 - Direct mapping between EWD Custom tags and ExtJS Javascript widget classes

EWD and ExtJS

- Example...

EWD and ExtJS

- How much simpler does EWD make ExtJS-based development?

Example: Ext-JS Grid

Design-focussed Development: ExtJS Grid Examples - Windows Internet Explorer

http://127.0.0.1/ewdphp/extjs/AjaxWorldDemo1.php

File Edit View Favorites Tools Help

Design-focussed Development: ExtJS Grid Examples

Array Grid

Company	Price	Change	% Change	Last Updated
3m Co	71.72	0.02	0.03	9/1 12:00am
Alcoa Inc	29.01	0.42	1.47	9/1 12:00am
Altria Group Inc	83.81	0.28	0.34	9/1 12:00am
American Express Company	52.55	0.01	0.02	9/1 12:00am
American International Group Inc.	64.13	0.31	0.49	9/1 12:00am
AT&T Inc.	31.61	-0.48	-1.54	9/1 12:00am
Boeing Co.	75.43	0.53	0.71	9/1 12:00am
Caterpillar Inc.	67.27	0.92	1.39	9/1 12:00am
Citigroup Inc.	49.37	0.02	0.04	9/1 12:00am
E.I. du Pont de Nemours and Company	40.48	0.51	1.28	9/1 12:00am
Exxon Mobil Corp	68.1	-0.43	-0.64	9/1 12:00am
General Electric Company	34.14	-0.08	-0.23	9/1 12:00am
General Motors Corporation	30.27	1.09	3.74	9/1 12:00am
Hewlett-Packard Co.	36.53	-0.03	-0.08	9/1 12:00am

Done Internet 100%

Ext-JS Javascript version

Design-focussed Development: ExtJS Grid Examples - Windows Internet Explorer

http://127.0.0.1/ewdphp/extjs/AjaxWorldDemo1.php

File Edit View Favorites Tools Help

Design-focussed Development: ExtJS Grid Examples

Array Grid

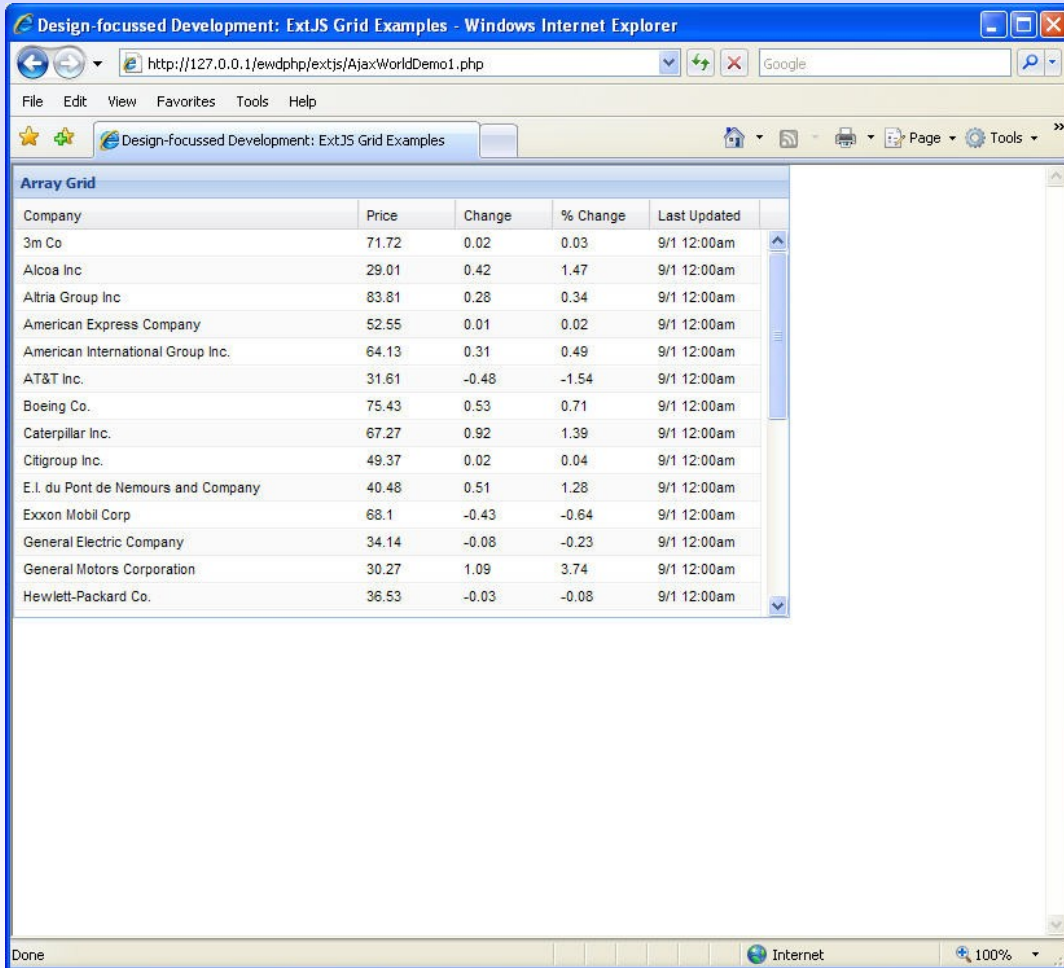
Company	Price	Change	% Change	Last Update
3m Co	71.72	0.02	0.03	9/1 12:00a
Alcoa Inc	29.01	0.42	1.47	9/1 12:00a
Altria Group Inc	83.81	0.28	0.34	9/1 12:00a
American Express Company	52.55	0.01	0.02	9/1 12:00a
American International Group Inc.	64.13	0.31	0.49	9/1 12:00a
AT&T Inc.	31.61	-0.48	-1.54	9/1 12:00a
Boeing Co.	75.43	0.53	0.71	9/1 12:00a
Caterpillar Inc.	67.27	0.92	1.39	9/1 12:00a
Citigroup Inc.	49.37	0.02	0.04	9/1 12:00a
E.I. du Pont de Nemours and Company	40.48	0.51	1.28	9/1 12:00a
Exxon Mobil Corp	68.1	-0.43	-0.64	9/1 12:00a
General Electric Company	34.14	-0.08	-0.23	9/1 12:00a
General Motors Corporation	30.27	1.09	3.74	9/1 12:00a
Hewlett-Packard Co.	36.53	-0.03	-0.08	9/1 12:00a

Done

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Array Grid Example</title>
<link rel="stylesheet" type="text/css" href="../../resources/css/ext-all.css" />
<link rel="stylesheet" type="text/css" href="../../resources/css/xtheme-default.css" />
<script type="text/javascript" src="../../adapter/ext/ext-base.js"></script>
<script type="text/javascript" src="../../ext-all.js"></script>
<script type="text/javascript" src="array-grid.js"></script>
<link rel="stylesheet" type="text/css" href="grid-examples.css" />
<link rel="stylesheet" type="text/css" href="../shared/examples.css" />
<script language="javascript">
Ext.onReady(function(){

    var store = new Ext.data.SimpleStore({
        fields: [
            {name: 'company'},
            {name: 'price', type: 'float'},
            {name: 'change', type: 'float'},
            {name: 'pctChange', type: 'float'},
            {name: 'lastChange', type: 'date', dateFormat: 'n/j h:ia'}
        ]
    });
    store.loadData(myData);
    var grid = new Ext.grid.GridPanel({
        store: store,
        columns: [
            {id: 'company', header: "Company", width: 160, sortable: true, dataIndex: 'company'},
            {header: "Price", width: 75, sortable: true, renderer: 'usMoney', dataIndex: 'price'},
            {header: "Change", width: 75, sortable: true, renderer: change, dataIndex: 'change'},
            {header: "% Change", width: 75, sortable: true, renderer: pctChange, dataIndex: 'pctChange'},
            {header: "Last Updated", width: 85, sortable: true, renderer: Ext.util.Format.dateRenderer('m/d/Y'), dataIndex: 'lastChange'}
        ],
        stripeRows: true,
        autoExpandColumn: 'company',
        height: 350,
        width: 600,
        title: 'Array Grid'
    });
    grid.render('grid-example');
    grid.getSelectionModel().selectFirstRow();
});
</script>
</head>
<body>
<h1>Array Grid Example</h1>
<div id="grid-example"></div>
</body>
</html>
```

Ext-GWT (Java) version

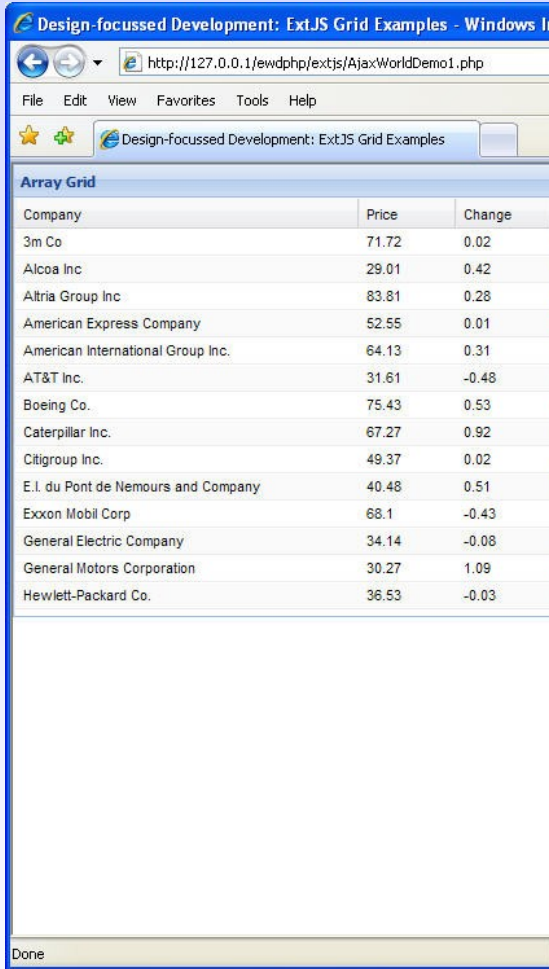


The screenshot shows a Windows Internet Explorer browser window titled "Design-focussed Development: ExtJS Grid Examples". The address bar shows the URL "http://127.0.0.1/ewdphp/extjs/AjaxWorldDemo1.php". The browser displays a table titled "Array Grid" with the following data:

Company	Price	Change	% Change	Last Updated
3m Co	71.72	0.02	0.03	9/1 12:00am
Alcoa Inc	29.01	0.42	1.47	9/1 12:00am
Altria Group Inc	83.81	0.28	0.34	9/1 12:00am
American Express Company	52.55	0.01	0.02	9/1 12:00am
American International Group Inc.	64.13	0.31	0.49	9/1 12:00am
AT&T Inc.	31.61	-0.48	-1.54	9/1 12:00am
Boeing Co.	75.43	0.53	0.71	9/1 12:00am
Caterpillar Inc.	67.27	0.92	1.39	9/1 12:00am
Citigroup Inc.	49.37	0.02	0.04	9/1 12:00am
E.I. du Pont de Nemours and Company	40.48	0.51	1.28	9/1 12:00am
Exxon Mobil Corp	68.1	-0.43	-0.64	9/1 12:00am
General Electric Company	34.14	-0.08	-0.23	9/1 12:00am
General Motors Corporation	30.27	1.09	3.74	9/1 12:00am
Hewlett-Packard Co.	36.53	-0.03	-0.08	9/1 12:00am

The browser status bar at the bottom shows "Done", "Internet", and "100%".

Ext-GWT (Java) version



Design-focussed Development: ExtJS Grid Examples - Windows I

http://127.0.0.1/ewdphp/extjs/AjaxWorldDemo1.php

File Edit View Favorites Tools Help

Design-focussed Development: ExtJS Grid Examples

Array Grid

Company	Price	Change
3m Co	71.72	0.02
Alcoa Inc	29.01	0.42
Altria Group Inc	83.81	0.28
American Express Company	52.55	0.01
American International Group Inc.	64.13	0.31
AT&T Inc.	31.61	-0.48
Boeing Co.	75.43	0.53
Caterpillar Inc.	67.27	0.92
Citigroup Inc.	49.37	0.02
E.I. du Pont de Nemours and Company	40.48	0.51
Exxon Mobil Corp	68.1	-0.43
General Electric Company	34.14	-0.08
General Motors Corporation	30.27	1.09
Hewlett-Packard Co.	36.53	-0.03

Done

```
column = new Column<Stock>(
column.setId("symbol");
column.setHeader("Symbol");
column.setWidth(150);
configs.add(column);
```

```
column = new Column<Stock>(
column.setId("last");
column.setHeader("Last Price");
column.setAlignment("center");
column.setWidth(70);
column.setRenderer(new NumberCellRenderer());
configs.add(column);
```

```
column = new Column<Stock>(
column.setAlignment("center");
column.setRenderer(new NumberCellRenderer());
configs.add(column);
```

```
column = new Column<Stock>(
column.setAlignment("center");
column.setDateFormatter(new DateFormatter());
configs.add(column);
```

```
ListStore<Stock> store = new ListStore<Stock>(
store.add(TestData.STOCKS);
```

```
ColumnModel cm = new ColumnModel(
cm.addConfig(column);
```

```
ContentPanel cp = new ContentPanel(
cp.setBorder("basic");
cp.setHeader("Basic Example");
cp.setAlignment("center");
cp.setLayout(new BorderLayout());
cp.setSize(600, 300);
```

```
Grid<Stock> grid = new Grid(
grid.setStyleAttribute("width", "100%");
grid.setAutoExpand();
grid.setBorders(true);
cp.add(grid);
```

```
add(cp);
}
```

```
}
```

```
package com.extjs.gxt.samples.client.examples.grid;
```

```
import java.util.ArrayList;
import java.util.List;
```

```
import com.extjs.gxt.samples.resources.client.TestData;
import com.extjs.gxt.samples.resources.client.model.Stock;
import com.extjs.gxt.ui.client.Style.HorizontalAlignment;
import com.extjs.gxt.ui.client.store.ListStore;
import com.extjs.gxt.ui.client.widget.ContentPanel;
import com.extjs.gxt.ui.client.widget.LayoutContainer;
import com.extjs.gxt.ui.client.widget.grid.ColumnConfig;
import com.extjs.gxt.ui.client.widget.grid.ColumnData;
import com.extjs.gxt.ui.client.widget.grid.ColumnModel;
import com.extjs.gxt.ui.client.widget.grid.Grid;
import com.extjs.gxt.ui.client.widget.grid.GridCellRenderer;
import com.extjs.gxt.ui.client.widget.layout.FitLayout;
import com.extjs.gxt.ui.client.widget.layout.FlowLayout;
import com.extjs.gxt.ui.client.widget.table.NumberCellRenderer;
import com.google.gwt.i18n.client.DateTimeFormat;
import com.google.gwt.i18n.client.NumberFormat;
```

```
public class GridExample extends LayoutContainer {
```

```
public GridExample() {
setLayout(new FlowLayout(10));
```

```
final NumberFormat currency = NumberFormat.getCurrencyFormat();
final NumberFormat number = NumberFormat.getFormat("0.00");
final NumberCellRenderer<Grid<Stock>> numberRenderer = new NumberCellRenderer<Grid<Stock>>(
currency);
```

```
GridCellRenderer<Stock> change = new GridCellRenderer<Stock>() {
public String render(Stock model, String property, ColumnData config, int rowIndex,
int colIndex, ListStore<Stock> store) {
double val = (Double)model.get(property);
String style = val < 0 ? "red" : "green";
return "<span style='color:" + style + "'>" + number.format(val) + "</span>";
}
};
```

```
GridCellRenderer<Stock> gridNumber = new GridCellRenderer<Stock>() {
public String render(Stock model, String property, ColumnData config, int rowIndex,
int colIndex, ListStore<Stock> store) {
return numberRenderer.render(null, property, model.get(property));
}
};
```

```
List<ColumnConfig> configs = new ArrayList<ColumnConfig>();
```

```
ColumnConfig column = new ColumnConfig();
column.setId("name");
column.setHeader("Company");
column.setWidth(200);
configs.add(column);
```

Design-focused version

Company	Price	Change	% Change	Last Updated
3m Co	71.72	0.02	0.03	9/1 12:00am
Alcoa Inc	29.01	0.42	1.47	9/1 12:00am
Altria Group Inc	83.81	0.28	0.34	9/1 12:00am
American Express Company	52.55	0.01	0.02	9/1 12:00am
American International Group Inc.	64.13	0.31	0.48	9/1 12:00am
AT&T Inc.	31.61	-0.48	-1.52	9/1 12:00am
Boeing Co.	75.43	0.53	0.70	9/1 12:00am
Caterpillar Inc.	67.27	0.92	1.37	9/1 12:00am
Citigroup Inc.	49.37	0.02	0.04	9/1 12:00am
E.I. du Pont de Nemours and Company	40.48	0.51	1.26	9/1 12:00am
Exxon Mobil Corp	68.1	-0.43	-0.63	9/1 12:00am
General Electric Company	34.14	-0.08	-0.23	9/1 12:00am
General Motors Corporation	30.27	1.09	3.60	9/1 12:00am
Hewlett-Packard Co.	36.53	-0.03	-0.08	9/1 12:00am

```
<ewd:config fetchMethod="ewd.test.getAJWGridData">
<ext:config path="/ext-2.2" />

<html>
<head>
<title>Design-focused Development: ExtJS Grid Examples</title>
</head>
<body>

<ext:grid datastore="stocks" title="Array Grid" stripeRows="true" autofill="false" width="600" height="350">
<ext:gridColumn header="Company" width="265" sortable="true" />
<ext:gridColumn header="Price" type="float" width="75" sortable="true" />
<ext:gridColumn header="Change" type="float" width="75" sortable="true" />
<ext:gridColumn header="% Change" type="float" width="75" sortable="true" />
<ext:gridColumn header="Last Updated" width="85" type="date" dateFormat="n/j h:ia" sortable="true" />
</ext:grid>

</body>
</html>
```


The ExtJS Demo

- Index4.ewd
 - Viewport + sub-panels
- trialSelection.ewd
 - Grid
- demographicsForm3.ewd
 - Form panel + fields

Conclusions

- EWD is probably the fastest way to develop web/Ajax applications
 - No compromises
 - All the most advanced modern techniques are available
 - But can be used in a fraction of the time
- Design-focussed = low maintenance
- GT.M + EWD = an awesome and world-class environment

