jQuery Datatables
Part IV: A New Hope

Widget of the Week (WoW) Series
Things to know …

- DOM
  - Document Object Model
- jQuery
  - Third-party JavaScript library
- Progressive enhancement
  - More than just a buzzword. (Two buzzwords.)
- Datatables.net
  - jQuery plug-in
DOM

- “Document Object Model”
- Way to represent, and interact with, the parts of HTML/XHTML and XML documents
- Elements, attributes, parents, siblings, children
- Kinda messy to work with the raw API, which is why we use something like …
jQuery

- Arguably the most popular JavaScript library in use today
- Excels at DOM manipulation, event handling, client-server interaction
- Good cross-browser support
- Readily extensible through plug-ins and libraries …
Progressive enhancement

“A strategy for web design that … uses web technologies in a layered fashion that allows everyone to access the basic content and functionality of a web page, using any browser or Internet connection, while also providing those with better bandwidth or more advanced browser software an enhanced version of the page.”
Progressive enhancement

- Create the page using basic low-tech accessible HTML content
- Use JavaScript/CSS/etc. to “enhance” the page with extra functionality / bling
- Enhancements can be minor, or can amount to complete rewrite of page
- Purports to separate content from presentation
Progressive enhancement

- Often compared with “graceful degradation”
  - Former assumes forward compatibility
  - Latter implicitly relies on backwards compatibility
Graceful degradation

- “Fault-tolerance” for the web
- Design pages for current standards
  - Accessibility
  - Previous generation platforms
- Assume “forward compatibility” of existing web technologies
Progressive enhancement

- Guarantees perpetual employment for client-side developers?
Use the 5Ps approach:

- Prepare
- Plan
- Prototype
- Plumb
- Proselytize
- Secure
Prepare

- Which widget(s) do I want/need to use?
  - Datatables.net plugin
- What docs are available?
  - jQuery online docs, uxt.ucsd.edu, datatables.net
- What client-side resources are required?
  - jQuery core, datatables plugin, HTML table
- What server-side resources are required?
  - Sample data source for dynamic example only
Plan

- How to generate table data?
  - Output from server side
  - Output on client side (AJAX-ish)
- First example: Generating the table on the server side (progressive enhancement)
- Second example: Build table dynamically
Prototype

- Create a basic HTML table and play around with the datatables.net plug-in.
<table>
<thead>
<tr>
<th>Reservoir</th>
<th>Capacity</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shasta</td>
<td>4,552,000</td>
<td>3,444,300</td>
</tr>
</tbody>
</table>

...
Plumb

- Create sample data source and skeleton table on page
- Add JavaScript glue code to write server-side data to page using DataTables API
Proselytize

» Tell your friends!
Secure

- How do you keep unauthorized users from gaining access to your services?
  - Use Jlink Security Filter to control access by role
  - Roll (role?) your own session-based authentication
  - Use Spring Security (a subject for another lunch hour)
Exercise IV

Create static HTML table and “enhance” with jQuery DataTables

› Optional: Add mouse-over highlighting
› Optional: Add click handler to cells
Exercise V

- What’s happening behind the scenes?
  - Examine in Firebug
  - Examine pageable table in Firebug
I AM YOUR FATHER!
Exercise V

- What are the implications?
  - Can we write data directly to the DOM?
  - Can we embed forms in pageable tables?
Exercise VI

- Dynamically generated table
- Optional: Dynamically generated column data
Clone Wars

- Yahoo! UI data table component
- jQuery Tablesorter
- jQueryTreeTable
Middleware, UI, and database
Coming soon
Datatables Part I: “The Phantom Menace”