

JavaScript Frameworks



Profound Logic Software

Alex Roytman



About Me

- I create Web Application Development and Modernization Tools for IBM i
- Have been doing it for 13 years
- Only IBM i vendor in this space that has chosen to build a JavaScript UI framework from the ground up specifically for IBM i





Why learn about Frameworks

- JavaScript frameworks are the wave of any future Web & Mobile application development
- Very important for developers to be aware and familiar with the concepts



What are JavaScript frameworks?

- Collection of 3rd party code
- Dropped into your Web project
- Shipped as some JavaScript files and often some CSS files



Building Blocks of Web Applications

- HTML
- CSS
- JavaScript
- Server-side language



HTML

- Not a programming language
- Tag based language
 <table>
 <input type="button">
- The “what is it” or “content” portion of the application



CSS (Cascading Style Sheets)

- The “how it looks” portion of your application
- Describes colors, shapes, borders, etc.
- Provides rules based on selectors



JavaScript

- Adds interactivity
- A high level programming language
- De facto language of browsers
- Can control the browser
- Can control how the browser reacts to the user
- Provides DHTML capabilities



A server-side language

- Provides access to data and business logic
- We will briefly look at RPG and PHP today
- Can be used in 2 basic ways:
 - Generate HTML and fill it with dynamic data
 - Generate just the data and pass it along to JavaScript (AJAX)





What frameworks provide

- Commonly used functions or API
- Extensions to JavaScript itself
- Widgets
- Built-in AJAX capabilities
- DOM Manipulation



Advantages

- The framework does the “heavy lifting” for common tasks
- No need to “reinvent the wheel”
- Resolves cross-browser issues
- Enforces consistency among developers and different parts of the application





Advantages

- Going beyond capabilities of HTML & CSS
 - Drag and drop
 - Dialogs
 - Cursor positioning / cursor sensitive functionality
 - Database-driven auto-suggest
 - Animation



Advantages

- Extend HTML & CSS with new Widgets
 - Panels
 - Pop-up calendar
 - Combo boxes
 - Tabs
 - Spinners
 - Sliders
 - Charts
 - Menus



Advantages

- Listing data in grids
 - Client-side or server-side sorting
 - Movable columns, Resizable columns
 - Row selection
 - Context menus
 - Paging bars
 - Built-in export to excel
 - Expanding rows
 - Single row zoom



Types of frameworks

- Frameworks to simply extend JavaScript and add simple DOM manipulation
 - Working with a set of API
 - Use selectors for DOM objects
 - Still creating most content with HTML
 - Can work alongside other frameworks
 - Example: jQuery



Types of frameworks

- UI Frameworks
 - Eliminate HTML coding altogether
 - Provide encapsulated widgets
 - Configuration with JSON in addition to API (combining the two)
 - Example: ExtJS



Configuration (JSON)

- framework.createForm (

```
{
  "width": "95%"
  "elements": [
    {
      "type": "textbox",
      "id": "OrdNum",
      "label": "Order Number"
    },
    {
      "type": "dropdown",
      "id": "Qty",
      "options": [ 10, 25, 50, 75, 100 ],
      "label": "Qty"
    },
    {
      "type": "button",
      "id": "Submit",
      "text": "Submit Order"
    }
  ]
}
```

);

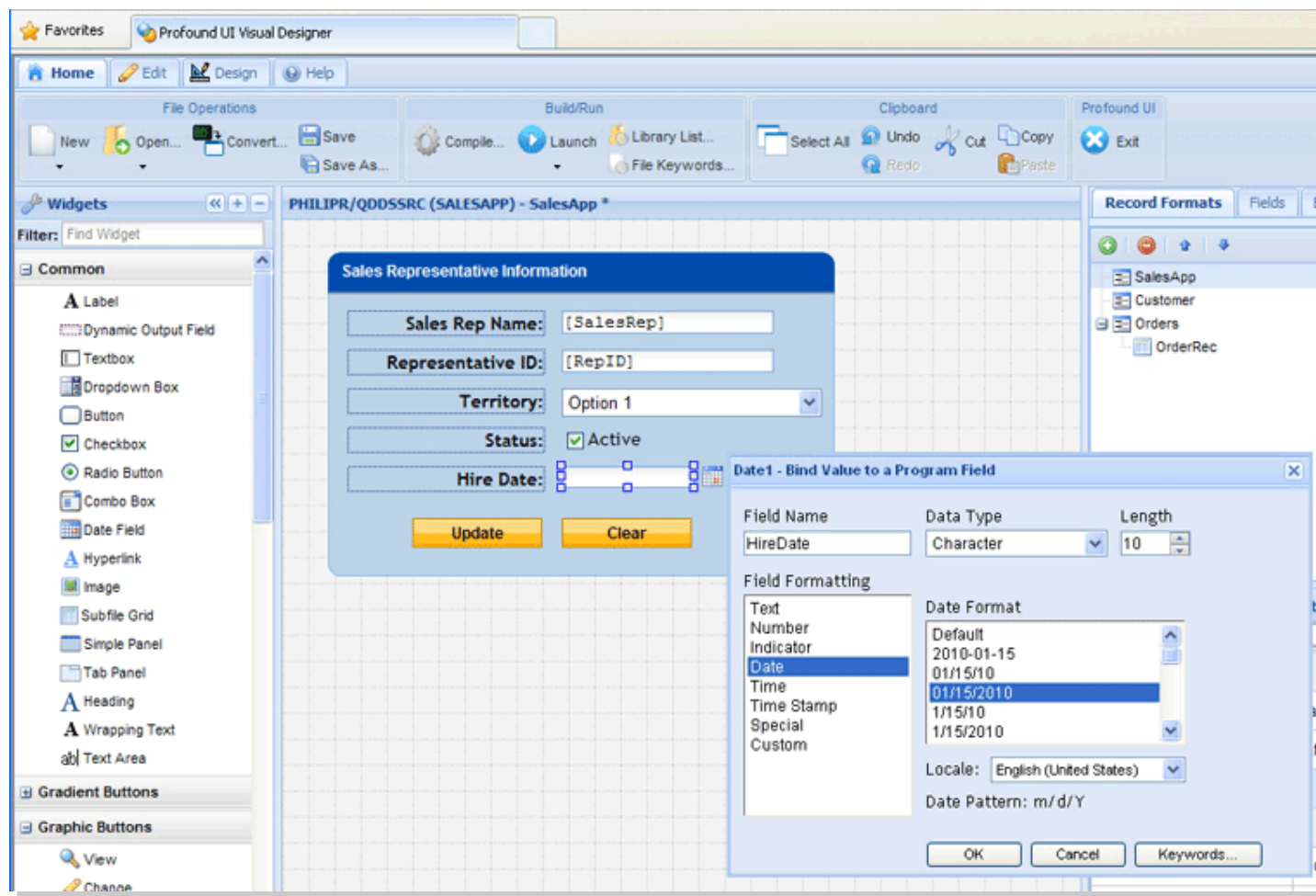


Types of frameworks

- JSON-based
 - UI definition is external to all of your code
 - Still provide JavaScript API, but the API are not necessary to build applications
 - Emphasis is on allowing to concentrate on business logic and server-side coding
 - Design tools helps create the JSON definition
 - Example: Profound UI



Design Tool





Creates .json file

```
{
  "width": "95%"
  "elements": [
    {
      "type": "textbox",
      "id": "OrdNum",
      "label": "Order Number"
    },
    {
      "type": "dropdown",
      "id": "Qty",
      "options": [ 10, 25, 50, 75, 100 ],
      "label": "Qty"
    },
    {
      "type": "button",
      "id": "Submit",
      "text": "Submit Order"
    }
  ]
}
```



JSON in DDS for RPG OA

- What is RPG Open Access
 - Allows a HANDLER to override I/O operations like READ, WRITE, and EXFMT
 - Overriding can mean providing a Web interface
 - HTML too simple to represent the interface – JSON is used to provide a Rich UI
- DDS is necessary to define the record



- ```
A R SEARCHFMT
A
A HTML('{"screen":{"record format nam-
A e":"SEARCHFMT","description":"Order-
A Search"},"items":[{"id":"Panel1","-
A field type":"panel","value":"Search-
A Orders","left":"20px","top":"20px"-
A ,"width":"445px","height":"250px","-
A panel style":"Simple","z index":"8"-
A ,"color":"#ffffff","text align":"ce-
A nter"}, {"id":"OutputField1","field -
A type":"output field","value":"Start-
A ing Date:","left":"40px","top":"65p-
A x","font weight":"bold","text align-
```



# Example - jQuery

- One of the most popular frameworks
- Open Source
- Small footprint
- Overloads \$
  - `$()` is the main jQuery function / object
  - Primarily used a selector
  - Kind of cryptic?
  - Does way too much?
- Uses chaining



# jQuery selectors & chaining

- Selector examples:
  - “#account”
  - “.numeric”
  - “input”
  - “:odd”
- Chaining example:
  - `$("#item").parents(".dialog").animate({ backgroundColor: "#fff568" }, "fast").animate({ backgroundColor: "#ffffff" }, "slow").addClass("deleted");`



# Thank You

**For comments/questions, please send to:**

**Alex Roytman**

[aroytman@profoundlogic.com](mailto:aroytman@profoundlogic.com)

|                     |                                                                  |
|---------------------|------------------------------------------------------------------|
| <b>Visit us at:</b> | <a href="http://www.profoundlogic.com">www.profoundlogic.com</a> |
| <b>Visit us at:</b> | <a href="http://www.profoundui.com">www.profoundui.com</a>       |
| <b>Call us at:</b>  | 1-877-224-7768                                                   |

