

## Flux, REST, and Javascript

### *Designing Flux Consoles to Target Unique User Needs*

**Reading Time: 3 minutes**

Customers rely on Flux to flexibly address their file orchestration and process automation challenges. A common requirement is the need to develop custom user interactions with Flux for targeted user communities or specific needs. Flux' s REST API provides an excellent starting point for discovering and exploiting Flux functionality to meet user needs without requiring a significant development effort and a lengthy development cycle.

Flux has developed a REST console application to demonstrate one such custom console. This code can be modified or enhanced to address particular or unique customer needs. The provided code is not a production ready application. At a minimum additional validity checking and exception handling would normally be added.

The REST console application is client side Javascript and HTML and runs within the user's browser. The development cycle consists of editing the code and refreshing your browser to pick up the changes. The HTML pages for the console are rendered largely using Bootstrap and JQuery. The REST console relies on the Flux REST API to interact with the Flux platform. The application uses Javascript libraries from JQuery, Bootstrap, Knockout, and MetaWidget. As delivered, the console fetches these libraries from CDN locations on the internet. When deployed into a customer environment these libraries should be downloaded and placed into the Flux webapp resources directory. The REST console application itself is deployed in the Flux webapp directory, and is accessed at (for example) `http://FluxServer:7186/restConsole.html`.

*Flux has developed a REST console application*

The REST console is a single file containing HTML and JavaScript. The application itself follows the MVVM (Model, View, View-Model) design pattern. The HTML contains a div for each section (or view) of the console. There are views to login, configure and submit workflows, review the workflow repository, the audit logs, workflows running on the engine, and the run history. Each div maps to a corresponding view model class written in Javascript. Each view model class defines a set of fields local to the class or observable by the div. Bootstrap is used to define observable fields that bind HTML fields to the Javascript view model class, minimizing the amount of code that needs to be written. Any time an observable field is changed in the view model class, the corresponding field in the HTML div is updated. The model consists of data retrieved from Flux using Ajax requests to Flux's REST API.

The login view model provides for basic authentication, signs in to a secured Flux engine, and then calls the populate method on each view model class. Each populate method makes a

JQuery Ajax call to a Flux API to retrieve the required data to populate that view model's bound div. The configure view model uses Metawidget to retrieve a json form definition from the workflow description field. The json form is used to request workflow variables from the user to populate the workflow before doing a JQuery Ajax submit to Flux using a namespace entered into the form. The other view models provide views on Flux's audit logs, run history, repository workflows and running workflows.

*One can construct a diverse set of capabilities to tailor Flux to your needs.*

Using this code as an example and through further exploring the REST API for Flux, one can construct a diverse set of tools and capabilities to augment and tailor Flux to your needs. If desired, Flux services staff are available on a billable services agreement to construct consoles to your specific requirements.

See the following for additional information regarding code and libraries used in the REST console application.

Code Library	Library URL
Flux REST API	<a href="http://support.flux.ly/80/restapi/">http://support.flux.ly/80/restapi/</a>
Bootstrap	<a href="http://getbootstrap.com">http://getbootstrap.com</a>
JQuery	<a href="http://jquery.com">http://jquery.com</a>
Knockout	<a href="http://knockoutjs.com">http://knockoutjs.com</a>
Metawidget	<a href="http://metawidget.org">http://metawidget.org</a>

### *About Flux*

The Flux software platform orchestrates file transfers and batch processing workflows for banking and finance. First released in 2000, Flux has grown into a financial platform that the largest US, UK, and Canadian banks and financial services organizations rely on daily for their mission critical financial systems.

### *Contact Flux*

+1 702-789-0907  
 sales@flux.ly  
 www.flux.ly