ArcGIS Workflow Manager JavaScript Viewer

Overview:

The Workflow Manager JavaScript Viewer is a fully functional sample application used as is with basic configuration. Configuration and deployment steps are provided in this document. Basic knowledge of ArcGIS Server / ArcGIS Portal, and Workflow Manager Server REST services is assumed.

Supported Browsers:

- Chrome
- Firefox
- Internet Explorer 10+

Deployment:

Execute the following steps to deploy the application:

1. Install an HTTP server such IIS or Apache.
2. Copy the contents of the Workflow Manager JavaScript viewer folder into the web folder of the HTTP server, either at the root or a subfolder. (e.g. C:\Inetpub\wwwroot, C:\Inetpub\wwwroot\wmviewer, C:\Apache\htdocs, C:\Apache\htdocs\wmviewer).
3. Modify the deployed **js/app/WorkflowManager/config/AppConfig.js** file to configure the Workflow Manager service, LOI Map service, ArcGIS Token service / ArcGIS Portal service if these are secured services (details below) and to configure custom basemaps (details below).
4. Modify the deployed **proxy/proxy.config** file to configure the Workflow Manager service that the proxy will forward to (details below).
5. If using IIS, create a website.
6. Launch the sample viewer in a web browser.

Configuration for AppConfig.js:

Several properties can be configured for the viewer. Modify the deployed js/app/WorkflowManager/config/AppConfig.js file.

Application settings

- **app.ServiceRoot** – Specifies the REST URL of the Workflow Manager Service.
- **app.AuthenticationMode** – Specifies whether to use ArcGIS Server Token authentication, ArcGIS Portal authentication, Windows authentication or no authentication.
  Note: ArcGIS Portal authentication is available in ArcGIS Workflow Manager Server 10.3.1 and later versions
- **app.TokenService** - Specifies the URL of the ArcGIS Token service. Applicable only if AuthenticationMode is set to token authentication.
- **app.PortalURL** - Specifies the URL of ArcGIS Portal. Applicable only if AuthenticationMode is set to portal authentication.
- **app.AppId** - Specifies the application ID of the viewer after being registered with ArcGIS Portal. Applicable only if AuthenticationMode is set to portal authentication

- **app.DefaultUser** – Specifies the default user name for the web application. Note that the user should be a valid Workflow Manager Application user.
- **app.AutoLogin** – Specifies whether to automatically login the default user when application is opened. Note that this applies only when no authentication is used and a valid Workflow Manager user is specified as default user.
- **app**.**DefaultQuery –** Specifies the default query at launch of the application using query path. The query path includes all the container names to indicate the hierarchy separated by backslash.
- **app.jobLOILayer –** Configuration for the dynamic Location of Interest (LOI) Map Service. The LOI Map Service is comprised of the Point of Interest (POI) feature class (optional) and Area of Interest (AOI) feature class with the feature classes being the top most layers.
  Note: Support for a new Point of Interest (POI) feature class was added in ArcGIS Workflow Manager Server 10.4.
  - **url** - Specifies the REST URL of the Location of Interest (LOI) Map service.
  - **POILayerID –** (optional) Layer ID of the Point of Interest (POI) feature layer within the map service.
  - **AOILayerID –** Layer ID of the Area of Interest (AOI) feature layer within the map service.
- **proxy.url** – Specifies the location of the proxy.
- **geometryServiceURL** – Specifies the location of the ArcGIS Geometry service.

Map settings

- **map.defaultBasemap** – Specifies the id of the basemap to use as the default basemap.
- **map.basemapGallery**
  - **isEnabled** – Specifies whether or not to enable a map gallery for selecting basemaps.
  - **showArcGISBasemaps** – Specifies whether or not to use ArcGIS Online basemaps in the basemap gallery. If this property is set to “false”, custom basemaps will need to be configured.
- **map.customBasemaps** – Specifies the basemaps to be displayed in the basemap gallery instead of ArcGIS Online basemaps.
- **map.initialExtent** - Specifies the application’s Area of Interest (AOI) map extent on log in.
- **map**.**search** – Specifies the application’s search controls for the map.
  - **zoomLevel** – Specifies the zoom scale to use for the result if there is not an associated zoom scale for the result.
  - **customSources** – Specifies whether or not to use default or custom locator sources.
  - **locatorSources** – Specifies the locator sources to use to find search results. Only applicable if using custom locator sources.
- **map.navigation** – Specifies the navigation controls for the map.
- **map.overview** – Specifies whether or not to include an overview map.
- **map.scalebar** – Specifies whether or not to include a scale bar in the map.
- **map.coordinates** – Specifies whether or not to show map coordinates.
- **map.legend** – Specifies whether or not to show map legend.
- **map.drawTool**
  - **isEnabled** – Specifies whether or not to show map draw tools.
  - **tools** – Specifies which draw tools to display.
  - **pointTolerance** – Selecting a point on the map is a little more difficult than selecting a polygon because the mouse click is required to fall directly on the point in order for a result to be returned. To makes things easier, a “tolerance” envelope can be built around the clicked point by specifying a point tolerance (number of pixels).

Configuration for proxy.config:

Configure the proxy settings for the application. Modify the deployed proxy/proxy.config file.

- **serverUrl** – Specifies the Workflow Manager Server REST endpoint. Change “myserver” to the hostname of your ArcGIS server.

Set up a localized viewer for supported languages:

Workflow Manager JavaScript Viewer supports the following additional languages:

- Chinese (Simplfied): zh-CN
- Italian: it
- Russian: ru
- Spanish: es
- Turkish: tr

Configure language support

1. Modify the deployed index.html file.
2. Remove ‘//’ (uncomment) from the locale you want to use with the application. Only one language can be configured at a time.
   Example: locale: "es",
3. Save index.html (or index.aspx).

Set up web security using Windows Authentication with IIS:

Web security can be set up for ArcGIS Workflow Manager Javascript viewer using Windows Authentication with IIS.

1. Enable Windows Authentication in IIS.
   For details on how to enable Windows Authentication in IIS, refer to Windows Authentication.
2. Make sure Anonymous Authentication is disabled for your website.
3. In the JavaScript viewer folder, rename the 'web.config' file to 'web-secure.config'.
4. Modify the deployed js/app/WorkflowManager/config/AppConfig.js file to use Windows Authentication.
   a. Set the Authentication mode to Windows
      AuthenticationMode: "windows"
5. Rename the file extension of index.html to index.aspx. This enables the file to be edited as a .NET file.
6. Open the index.aspx for editing.
   a. Enable the application to use the Windows Authenticated user
      i. For repositories without domain authentication remove ‘//’ (uncomment) from the line
         var user = "<%= User.Identity.Name.Replace("MyDomain\","") %>";
         Note: Replace ‘MyDomain’ to match the organization’s domain.
      ii. For domain authentication remove ‘//’ (uncomment) from the line
         var user = "<%= User.Identity.Name.Replace("\","*") %>";
   b. Setup the application to use the Windows Authenticated user at startup.
      i. Remove ‘//’ (uncomment) from the line
         Controller.startup({user: user});

ii. Put '//' (comment) in the line
//Controller.startup();

Note: If you are unsure of the domain name, remove '//' (uncomment) from the following lines. The windows user returned will include the domain name along with the username. The domain name is case sensitive.

```
var windowsUser = "<%= User.Identity.Name %>";
var loadingMsgElem = document.getElementById("loadingMessage");
loadingMsgElem.innerHTML = "windowsUser = " + windowsUser;
```

7. Save index.aspx.
8. Open the JavaScript viewer using the url:
   http://your-server-name/website-name/index.aspx

   Note: Some web browsers automatically log users in. If not, enter the user login information and click OK.

Set up security using ArcGIS Portal Authentication:

ArcGIS Workflow Manager JavaScript viewer can be configured to use ArcGIS Portal authentication. Support for Portal authentication was added in ArcGIS Workflow Manager Server 10.3.1. This configuration requires ArcGIS Server to be federated with an on premise ArcGIS Portal installation.

- Federate ArcGIS Server with ArcGIS Portal using ArcGIS Web Adaptor.
  For details on how to federate ArcGIS Server, refer to the ArcGIS Portal topic "**Federate an ArcGIS Server site with your portal**".
- Install ArcGIS Workflow Manager Server on the federated ArcGIS Server site and publish your Workflow Manager service.

In ArcGIS Portal 10.5, the Workflow Manager JavaScript viewer can be deployed to a predefined location in the on premise Portal application directory. The application directory for the Workflow Manager web application already has a preregistered appId, so no additional web application registration is needed.

1. Copy the contents of the Workflow Manager JavaScript viewer folder into the Portal installation directory: **\ArcGIS\Portal\apps\workflow**
2. Modify the deployed **workflow\app\WorkflowManager\config\AppConfig.js** file to use ArcGIS Portal Authentication.
   a. Set the Authentication mode to Portal
      AuthenticationMode: "portal"
   b. Update the Portal URL
      PortalURL: "https://your-server/portal"
   c. Verify the AppId is set to "wmxweb".
3. To access the JavaScript viewer:
   a. Log into Portal and search for ArcGIS Workflow Manager application
   b. Access by URL: https://your-server.domain/portal/apps/workflow

For ArcGIS Portal versions earlier than 10.5, the Workflow Manager JavaScript viewer will need to be manually added and registered as an application in Portal.

1. Copy the contents of the Workflow Manager JavaScript viewer folder into the web folder of the HTTP server, either at the root or a subfolder. (e.g. C:\Inetpub\wwwroot, C:\Inetpub\wwwroot\wmviewer, C:\Apache\htdocs, C:\Apache\htdocs\wmviewer).
2. Add the JavaScript viewer as an item in ArcGIS Portal.
   For details on how to add item to ArcGIS Portal, refer to the ArcGIS Portal topic "**Add items**".
3. Register the JavaScript viewer with ArcGIS Portal.
   For details on how to register your application with ArcGIS Portal, refer to the ArcGIS Portal section "**Register your app**" in ArcGIS Portal topic "**Add items**".
4. In ArcGIS Portal Application **Settings**, update **App Registration** and add the machine hosting the JavaScript viewer to the existing Redirect URIs. The redirect URI can be redirected to after users successfully login.
   Redirect URI: https://[hostname].[domain]
5. Copy the application Id of the registered JavaScript viewer.
6. Modify the deployed **js/app/WorkflowManager/config/AppConfig.js** file to use ArcGIS Portal Authentication.
   a. Set the Authentication mode to Portal
      AuthenticationMode: "portal"
   b. Update the Portal URL
      PortalURL: "https://your-server/portal"
   c. Update the Application Id of the registered JavaScript viewer. This is the same application Id copied from Portal in a previous step.
      AppId: "your-application-id"
7. To access the JavaScript viewer:
   a. Log into Portal and click on the Workflow Manager JavaScript viewer registered in Portal
   b. Access directly by the application URL

Using Token Authentication

When using ArcGIS Server Token authentication with Server built-in roles and users from the Windows domain, you need to provide the domain along with the username to login into the viewer. This is irrespective of whether domains are enabled for use in Workflow Manager.

Deployment in Disconnected Environments

When deploying the JavaScript Viewer in a disconnected environment where there is no Internet connection available or Internet access is prohibited by your organization, referenced sources need to be replaced with sources available locally.

The items that you'll need to repoint to local resources include:

- ArcGIS JavaScript API
- ArcGIS Services
  - Maps
  - Geometry Service
  - Locator Services
- Third Party Software – Font Awesome

Note: Refer to the ArcGIS Portal topic "Configure a disconnected deployment" for details on how to configure the resources and services in disconnected portal.

ArcGIS JavaScript API

Download and install the ArcGIS JavaScript API.

1. Determine the version of the JavaScript API used by the application from the index.html file.
2. Download the appropriate version of the JavaScript API from https://developers.arcgis.com/downloads/ and deploy to your web server.
   Note: If deploying the JavaScript Viewer to ArcGIS Portal, the JavaScript API comes preinstalled at /portal/framework/jsapi/.
3. Modify the deployed **index.html** file.
   a. Replace the following references with locally installed versions.
   <link rel="stylesheet" href="//js.arcgis.com/[version]/esri/css/esri.css">
   <script type="text/javascript" src="//js.arcgis.com//[version]/"></script>

   e.g.
   <link rel="stylesheet" href="//[your-server].[domain]/jsapi/esri/css/esri.css">
   <script type="text/javascript" src="//[your-server].[domain]/jsapi/"></script>
   Or
   <link rel="stylesheet" href="//[your-server].[domain]/portal/framework/jsapi/jsapi/esri/css/esri.css">
   <script type="text/javascript" src="//[your-server].[domain]/portal/framework/jsapi/jsapi/"></script>

ArcGIS Services

Modify the deployed **js/app/WorkflowManager/config/AppConfig.js** file to reference local services.

1. Map Services
   Update the map section with your offline maps. Refer to the BasemapLayer class for supported types which include tiled base maps (https://developers.arcgis.com/javascript/3/jsapi/basemaplayer-amd.html).
   Note: Support for dynamic basemaps will be included in a future release.
   a. **map.defaultBasemap** - Update with your custom basemap name
   b. **map.basemapGallery**
      i. **isEnabled** – Specifies whether or not to enable a map gallery for selecting basemaps.
      ii. **showArcGISBasemaps** – Set to “false”
   c. **map.customBasemaps** - Update to your local basemap services. Remove streets, hybrid and topo basemaps which are references to ArcGIS Online services.
   e.g.

```
customBasemaps : [
 {
  id: “localbasemap",
  title: "localbasemap",
  layers: [{
   url: "https://[hostname].[domain]/arcgis/rest/services/[localbasemap]/MapServer"
  }],
  thumbnailUrl: "js/widget/gis/BasemapGallery/images/localbasemap.jpg"
 }
]
```

2. Geometry Service

a. **geometryServiceURL** – Update to local geometry service installed with ArcGIS Server
   e.g.
   http://[hostname].[domain]/arcgis/rest/services/Utilities/Geometry/GeometryServer

3. Locator Services
   The default locator service is configured to use ArcGIS Online and will not be available for disconnected deployments. You will need to provide your own locator services in a disconnected deployment.
   a. **locatorSources** - [your customized locator services]
      To customize locator sources, refer to
      https://developers.arcgis.com/javascript/3/jsapi/search-amd.html#sources.

Third Party Software – Font Awesome

The JavaScript Viewer makes use of the online Font Awesome font and css toolkit. The toolkit will need to be downloaded and deployed locally for disconnected environments.

1. Download FontAwesome version 3.2.1 from http://fontawesome.io/3.2.1/. Unzip and copy the folder to the deployment directory of your web application.
   e.g. C:\Inetpub\wwwroot\wmviewer\font-awesome
   \ArcGIS\Portal\apps\workflow\font-awesome
2. Modify the deployed **index.html** file.
   Replace the online font-awesome reference with the local location.

   ```
   <link rel="stylesheet" type="text/css" href="//netdna.bootstrapcdn.com/font-awesome/3.2.1/css/font-awesome.css">
   ```

   e.g.

   ```
   <link rel="stylesheet" type="text/css"
   href="//[hostname].[domain]/wmviewer/workflow/font-awesome/css/font-awesome.css">
   <link rel="stylesheet" type="text/css"
   href="//[hostname].[domain]/portal/apps/workflow/font-awesome/css/font-awesome.css">
   ```