# K2 Javascript Service Broker: Template for Product Documentation

|  |  |
| --- | --- |
| **Project** | [project/JS broker type name] |
| **Dev/QA Contact** | [primary contact’s name and email] |
| **TFS IDs/Spec Link** | [list applicable TFS ids for specs, user stories and work items related to this service type, or alternatively provide a link to the specifications for this project] |

# About this document

This documentation template is used to gather information to assist the K2 product documentation team to create documentation for a JavaScript (JS) Service Broker.

## What you need to do

Please fill out this template by reviewing the sections and replacing the highlighted areas with information relevant to your broker type, save the document with a unique name, and then send it to your K2 contact along with the rest of the files. Please also email this file to [documentation@k2.com](mailto:documentation@k2.com) so that they can prepare the required product documentation.   
Not all sections will be applicable to all JS Service Types. The more information you can provide in this document, the better the documentation team will be able to complete the documentation task without excessive technical review and other back-and-forth.

Contents

[About this document 1](#_Toc463421406)

[What you need to do 1](#_Toc463421407)

[Overview & Scenario 2](#_Toc463421408)

[Functionality 2](#_Toc463421409)

[Prerequisites 2](#_Toc463421410)

[Permissions 2](#_Toc463421411)

[Registering a Service Instance 2](#_Toc463421412)

[Runtime 2](#_Toc463421413)

[Versions Tested/Supported 2](#_Toc463421414)

[Service Instances 3](#_Toc463421415)

[Authentication Modes Supported 3](#_Toc463421416)

[Service Keys 3](#_Toc463421417)

[Service Instance Category Structure 3](#_Toc463421418)

[Service Objects, Properties and Methods 4](#_Toc463421419)

[SmartObjects, Properties and Methods 4](#_Toc463421420)

[Considerations 4](#_Toc463421421)

[Using the Service Type 4](#_Toc463421422)

[Troubleshooting 4](#_Toc463421423)

# Overview & Scenario

*Provide a brief overview of the integration that this JS service type enables. Include one or more scenarios that it is expected to be used. Example below:*

[The JS-Weather Service Type exposes the OpenWeather API as SmartObjects so that you can retrieve weather forecast details for a location]

# Functionality

*Provide a list of the standard functionality that is exposed by this type. Example below:*

Functionality provided:

* Retrieve (read) current weather for a specific location
* List forecast weather for a specific location

*If applicable and if known, provide a list of functionality that is NOT exposed by the type, but the end user might expect. This does not have to be an exhaustive or comprehensive list, but if there are standard things in the third-party data provider that are specifically not available in this type, list them. For example if there are specific endpoints/methods/objects that were deliberately left out of the broker, list them. Example below:*

Functionality not provided:

* Retrieve historical weather data

# Prerequisites

*If there are specific prerequisites for this JS type, describe them here. This could include licensing considerations for the third party technology (for example, the DocuSign type requires you to activate Developer mode in DocuSign and obtain a developer key from DocuSign), specific ports that may need to be opened to allow traffic to flow (for example, connecting to some provider might require a specific port to be opened), supporting components prerequisites (for example, your JS broker might require a supporting library.)*

# Permissions

*Describe specific permissions that may be required for this Service Type, breaking these into use cases if required. Basically, we are telling the user what permissions K2 needs to allow this type to work. Examples below:*

## Registering a Service Instance

* The account used to register an instance of the type requires the Administer permission on your OpenWeather subscription

## Runtime

* Accounts that call the OpenWeather API at runtime to retrieve data, require at least the Read permission in your openWeather subscription.
* Accounts that update data in OpenWeather requires at least Write permissions in your openWeather subscription

# Versions Tested/Supported

*If applicable, provide a list of the versions of the third-party technology that were tested with this type, or the versions that are expected to be supported in this type. Example below:*

* OpenWeather\_V1 API

# JS Broker Service Types and Service Instances

*State whether you should use the standard JS Broker Type/Instance forms to create this broker in a K2 environment, or whether the broker will be included with K2 by default, or if you need to use feature activation to get an instance of this JS broker. If there are specifics around registering an instance (for example that you have to use Feature Activation or a specific form to register a service instance, provide that information here. If there is a specific sequence of steps for registering a service type/instance, describe that here. If there is a Feature Activate page for this type, provide a screenshot of the feature activation page and describe any specifics for the feature activation page that the user has to know about.*

Add Instructions on how to register an instance of this JS type

## Authentication Modes Supported

*Not all services support all modes. List the Authentication Modes that are supported in this type. For reference, the available modes for JS brokers are listed below. For more, see* [*this topic*](http://help.k2.com/onlinehelp/k2blackpearl/UserGuide/4.7/default.htm#ServiceBrokers/AuthenticationModes/AuthenticationModes.htm) *in help. If there are specific considerations about Authentication Modes (for example, static credentials may require a specific format for the username), describe those considerations here.*

* OAuth
* Static

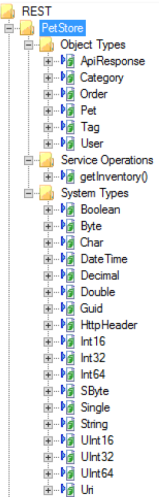
## Service Configuration Keys

*Explain each service instance configuration key, how it might affect the runtime behavior or service objects generated, and why you may want to use it. Also provide sample values for reference so that users can see what types of values are expected in the key (e.g. specific date/number/string/Boolean formats). Use a table structure; you can use the example below (sample values are provided for reference only; the rows in your table will be different):*

|  |  |  |  |
| --- | --- | --- | --- |
| **Key** | **Data type** | **Sample Value** | **Notes** |
| PrivateKey | Text | 123456-fgdhsltyrl011 | The key for your subscription (you can find this value at api.openweather.net) |
| Special Char Replacement | Text | \_ | This character will be used to replace any special characters in generated service object (e.g. spaces will be replaced with this character) |

## Service Instance Category Structure

*If your JS broker organizes artifacts into a specific structure, provide detail on what the different categories that may exist in an instance of the Service Type. In some cases this may be self-explanatory, but there may be instances where categories in the service instance are not obvious, such as Object Types, Service Operations. If the type exposes a specific category structure (e.g. endpoint types that always create an Object Types and Service Operations or SharePoint that exposes lists and libraries as folders) then add a screenshot showing the generated categories. The SmartObject tester tool is an easy way to grab this screenshot.*



## Service Objects, Properties and Methods

*If the Service always generates specific Service Objects, explain the purpose and use of these Service objects here. Describe the properties and methods on each service object to determine what types of functionality is exposed. This information will help designers who are creating SmartObjects that call Service object methods. If applicable, describe why you may not see all entities from the LOB (for example because certain data types are not supported, or a certain service key may restrict the Service Objects generated)*

|  |  |  |
| --- | --- | --- |
| Service Object Name | | |
| Properties | Name | Notes |
| [Property1 Name] | [Notes about the property, such a specific type or intended usage) |
| [Property2 Name] | [Notes about the property, such a specific type or intended usage) |
| … | … |
| Methods | Method | Notes |
| [Method1 Name] | [Notes about the method such as intended use, parameters and what the parameters do] |
| [Method2 Name] | [Notes about the method such as intended use, parameters and what the parameters do] |
| … | … |

## SmartObjects, Properties and Methods

*If the Service always generates specific SmartObjects, explain the purpose and use of these SmartObjects here. Describe the properties and methods on each SmartObject to determine what types of functionality is exposed. You may use a table or bulleted list.*

|  |  |  |
| --- | --- | --- |
| SmartObject Name | | |
| Properties | Name | Notes |
| [Property1 Name] | [Notes about the property, such a specific type or intended usage) |
| [Property2 Name] | [Notes about the property, such a specific type or intended usage) |
| … | … |
| Methods | Method | Notes |
| [Method1 Name] | [Notes about the method such as intended use, parameters and what the parameters do] |
| [Method2 Name] | [Notes about the method such as intended use, parameters and what the parameters do] |
| … | … |

# Considerations

*If there are other considerations about this service type, such as specific issues identified during testing, specific considerations about runtime use of the service, known issues, specific restrictions about how the service operates, please provide them here.*

* List specific considerations for this JS service type here

# Using the Service Type

*If there are specific ways you need to use the Service Type, the Service Objects or the Smart Objects in the service type, please describe them here. This could include something like a how-to sequence of steps, a sequence of how you may need to chain method calls in Service Object together when using the service type*

If applicable, describe how you could use the service type here, perhaps as a sequence of steps with screenshots

# Troubleshooting

*If there is any troubleshooting information or troubleshooting tips for this JS Type, add those notes here.*

If applicable, describe troubleshooting steps or tips for this service type

# Additional Notes

*Describe anything else you think will be important to note in the documentation for this service type.*

If necessary, you can add additional notes about the JS service type here