Last updated Dec 8, 2017   Rate this page: ☆ ☆ ☆ ☆ ☆

Developing gadgets

At heart, Atlassian gadgets are Google gadgets ↗ . Atlassian gadgets use the new Google `gadgets.*` API defined by the OpenSocial ↗ specification. In addition, you will probably want to use some of the Atlassian extensions to the gadget specification. You will also want to package your gadget as an Atlassian plugin, along with plugin modules that allow your gadget to interact with the Atlassian applications such as JIRA ↗ and Confluence ↗ .

Introducing Gadgets, Containers, Plugins and Applications

A gadget is essentially an XML file containing the gadget specification. You will use recognised XML elements to define the following:

1. Gadget characteristics, such as the author's name (your name), the gadget title and description, preferred sizing, etc.
2. A screenshot and/or thumbnail image that containers can display to show users what your gadget looks like.
3. Required features that containers must provide for your gadget.
4. User preferences, where your gadget can allow its users to customise certain aspects of the gadget display.
5. The content section, where you can define the content that your gadget will display. This is where you add the HTML and JavaScript functions that produce your gadget's output.

As the gadget developer, you create the static XML file and make it available on a server. The dashboard or other container will pull the XML file from the server or plugin where it resides, and render it. The container may, for example, display the gadget inside an iframe on a dashboard. Or the container may display the gadget on a web page via a wiki macro.

Here is an overview of how gadgets, containers, plugins and applications interact with each other:

- Google gadgets that support the OpenSocial ↗ specification will run on OpenSocial containers ↗ .
- A simple Google gadget will run on an Atlassian container, including the dashboard of any Atlassian application that supports gadgets.
- In principle, an Atlassian gadget will run on an OpenSocial container ↗ .
- You can customise your Atlassian gadget to interface in funky ways with your Atlassian applications, via the Atlassian Plugin Framework. See Packaging your Gadget as an Atlassian Plugin.

Writing a Simple Gadget

If you have never written a gadget before, you can start by writing a simple Google gadget:

1. Become familiar with Google gadgets. There are two Google gadget developer guides, one for the earlier 'legacy' version of the Google gadget API and one for the version which supports OpenSocial ↗ .
2. Take a look at our hints for developing in iGoogle sandbox, compiled from our own experiences with Google gadgets.
3. Read the guide to getting started with OpenSocial ↗ .

Writing an Atlassian Gadget

Now move on to write an Atlassian Gadget.

To write an Atlassian gadget, you can use the Atlassian extensions to the gadget specification:

1. Take a look at our example gadgets.
2. See the Atlassian gadget XML specification.
3. Deploy your gadget on a container.

For details, browse these topics:

- Creating your Gadget XML Specification
- Using Substitution Variables and Directives in your Gadget
- Allowing Configuration Options in your Gadget
- Including Features into your Gadget
- Gadget Containers
- Packaging your Gadget as an Atlassian Plugin
- Internationalising your Gadget
- Using Web Resources in your Gadget
- Using Atlassian REST APIs in your Gadget
- Providing User Authentication for Gadgets
- Using the Atlassian Gadgets JavaScript Framework

- Using the Atlassian Gadgets JavaScript Framework
- Managing Caching for your Gadget

Rate this page: ☆ ☆ ☆ ☆ ☆