

Include HTML content from other services

This is a live demo which uses content from Atlassian Market place status page

APIs **Operational**

90 days ago

100.0 % uptime

Today

Bitbucket Cloud APIs **Operational**

90 days ago

100.0 % uptime

Today

Confluence Cloud APIs **Operational**

90 days ago

100.0 % uptime

Today

Jira Cloud APIs **Operational**

90 days ago

100.0 % uptime

Today

Product Events ? **Operational**

90 days ago

100.0 % uptime

Today

User APIs **Operational**

90 days ago

100.0 % uptime

Today

Webhooks **Operational**

90 days ago

100.0 % uptime

Today

Web Triggers **Operational**

90 days ago

100.0 % uptime

Today

Developer **Operational**

90 days ago

99.9 % uptime

Today

App Deployment **Operational**

90 days ago

100.0 % uptime

Today

Artifactory (Maven repository) ? **Operational**

90 days ago

100.0 % uptime

Today

Create and manage apps ? **Operational**

90 days ago

100.0 % uptime

Today

Developer documentation ? **Operational**

90 days ago

100.0 % uptime

Today

Forge App Installation ? **Operational**

90 days ago

100.0 % uptime

Today

Forge CDN (Custom UI) ? **Operational**

90 days ago

100.0 % uptime

Today

Forge Function Invocation **Operational**

90 days ago

100.0 % uptime

Today

atlassian.com ? **Operational**

90 days ago

100.0 % uptime

Today

Forge App Logs ? **Operational**

90 days ago

100.0 % uptime

Today

Forge App Monitoring ? **Operational**

90 days ago

99.95 % uptime
Today
Developer console ? **Operational**
90 days ago
99.86 % uptime
Today
Forge direct app distribution ? **Operational**
90 days ago
100.0 % uptime
Today
Hosted storage ? **Operational**
90 days ago
98.67 % uptime
Today
Forge CLI **Operational**
90 days ago
100.0 % uptime
Today
End-user consent ? **Operational**
90 days ago
100.0 % uptime
Today
Forge App Alerts ? **Operational**
90 days ago
100.0 % uptime
Today
Marketplace ? **Under Maintenance**
90 days ago
99.91 % uptime
Today
App listing management ? **Operational**
90 days ago
100.0 % uptime
Today
App listings ? **Operational**
90 days ago
100.0 % uptime
Today
App pricing ? **Operational**
90 days ago
100.0 % uptime
Today
App submissions **Operational**
90 days ago
100.0 % uptime
Today
Category landing pages **Operational**
90 days ago
100.0 % uptime
Today
Evaluations and purchases **Operational**
90 days ago
100.0 % uptime
Today
In-product Marketplace and app installation (Cloud) ? **Operational**
90 days ago
100.0 % uptime
Today
In-product Marketplace and app installation (Server) ? **Operational**
90 days ago
100.0 % uptime
Today
Notifications ? **Operational**
90 days ago
100.0 % uptime
Today
Private listings **Operational**
90 days ago
99.06 % uptime
Today
Reporting APIs and dashboards ? **Under Maintenance**
90 days ago
99.76 % uptime
Today
Search **Operational**
90 days ago
100.0 % uptime
Today
Vendor management **Operational**

90 days ago
100.0 % uptime
Today
Vendor Home Page **Operational**
90 days ago
100.0 % uptime
Today
Authentication and user management **Operational**
90 days ago
99.71 % uptime
Today
Support **Operational**
90 days ago
100.0 % uptime
Today
Atlassian Support contact form **Operational**
Developer community ? **Operational**
90 days ago
100.0 % uptime
Today
Developer service desk ? **Operational**
Marketplace service desk ? **Operational**
Atlassian Support **Operational**
Vulnerability management [AMS] **Operational**
90 days ago
100.0 % uptime
Today

It uses a bit of a custom CSS and the configuration on the page looks like this:

CONF Doc ConfIDoc Excerpt Macro | serviceId = c577da46-df22-4...

HTML

```
<style type="text/css">
.component-status {
  color:green;
  font-weight:bold;
}
</style>
```

It all starts with Configuration definition and that looks like this:

Edit Configuration

Connection type:

Name:

The name for your configuration, will be visible to users

Service URL:

You can use placeholders for parameters which can be later set by users of this configuration (in macro editor).
Parameters should be wrapped in brackets (like []), should be alphanumeric (unless you want to urlencode it, see example below), no spaces allowed.

- `https://jira.atlassian.com/rest/api/latest/issue/{PARAM1}`
- `SELECT * FROM purchases WHERE id={MYPURCHASEID}`

If you need to UriEncode the parameter's value then use the following notation: `{(PARAM1)}`

Restrictions:

Comma separated user groups and/or users who should have access to this configuration
And will be able to use it using ConfiDoc macros

You can see it is very simple, no authentication is used and we just read the whole page from <http://status.marketplace.atlassian.com/>

Then we do some "magic" with selectors to select only the bits we are interested in

And the selector for ConfiDoc Excerpt macro looks like this

```
div class="components-container one-column"
```

We just take in the contents from this "DIV" element

This is how the ConfiDoc Excerpt macro looks like in the config mode

Edit 'ConfiDoc Excerpt Macro' Macro

use *

Services are defined by Confluence administrators through ConfiDoc plugin configuration UI

Include as *

Content type

CSS3-like Selector (html excerpts)

To include parts of the resource's content.
Example: `div id="myId"` or `div/class`

Marketplace Homepage Operational
Add-on Listing Pages Operational
Add-on Pricing Details Operational
Sales API Operational
Search Operational
UPM Connectivity Operational

With a bit of custom CSS we can make it more readable on our page (see live example on top of the page)

