



Google App Engine

Java Application Configuration

In addition to the [web.xml deployment descriptor](#), an App Engine Java application uses a configuration file, named `appengine-web.xml`, to specify the app's registered application ID and the version identifier of the latest code, and to identify which files in the app's WAR are static files (like images) and which are resource files used by the application. The [AppCfg](#) command uses this information when you upload the app.

- [About appengine-web.xml](#)
- [Static Files and Resource Files](#)
 - [Including and Excluding Files](#)
 - [Setting the Browser Cache Expiration for Static Files](#)
 - [Changing the MIME Type for Static Files](#)
 - [Changing the Root Directory](#)
- [System Properties and Environment Variables](#)
- [Configuring Secure URLs \(SSL\)](#)
- [Enabling Sessions](#)
- [Inbound Services](#)
- [Disabling Precompilation](#)
- [Administration Console Custom Pages](#)
- [Custom Error Responses](#)
- [Warming Requests](#)

About appengine-web.xml

An App Engine Java app must have a file named `appengine-web.xml` in its WAR, in the directory `WEB-INF/`. This is an XML file whose root element is `<appengine-web-app>`. A minimal file that specifies the application ID, a version identifier, and no static files or resource files looks like this:

```
<?xml version="1.0" encoding="utf-8"?>
<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
  <application>application-id</application>
  <version>1</version>
</appengine-web-app>
```

The `<application>` element contains the application's ID. This is the application ID you register when you create your application in [the Admin Console](#). When you upload your app, AppCfg gets the application ID from this file.

The `<version>` element contains the version identifier for the latest version of the app's code. The version identifier can contain letters, digits, and hyphens. AppCfg uses this version identifier when it uploads the application, telling App Engine to either create a new version of the app with the given identifier, or replace the version of the app with the given identifier if one already exists. You can test new versions of your app using a URL such as `http://version-id.latest.application-id.appspot.com`. You can select which version of the app your users see, the "default" version, using [the Admin Console](#).

The `<static-files>` and `<resource-files>` elements are described in the next section.

You can find the DTD and schema specifications for this file in the SDK's `docs/` directory.

Static Files and Resource Files

Many web applications have files that are served directly to the user's browser, such as images, CSS style sheets, or browser JavaScript code. These are known as *static files* because they do not change, and can benefit from web servers dedicated just to static content. App Engine serves static files from dedicated servers and caches that are separate from the application servers.

Files that are accessible by the application code using the filesystem are called *resource files*. These files are stored on the application servers with the app.

By default, all files in the WAR are treated as both static files and resource files, except for JSP files, which are compiled into servlet classes and mapped to URL paths, and files in the `WEB-INF/` directory, which are never served as static files and always available to the app as resource files.

Except as otherwise [noted](#), the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#).

Java is a registered trademark of Oracle and/or its affiliates

©2011 Google - [Code Home](#) - [Terms of Service](#) - [Privacy Policy](#) - [Site Directory](#)

Google Code offered in: [English](#) - [Español](#) - [日本語](#) - [한국어](#) - [Português](#) - [Русский](#) - [中文\(简体\)](#) - [中文\(繁體\)](#)