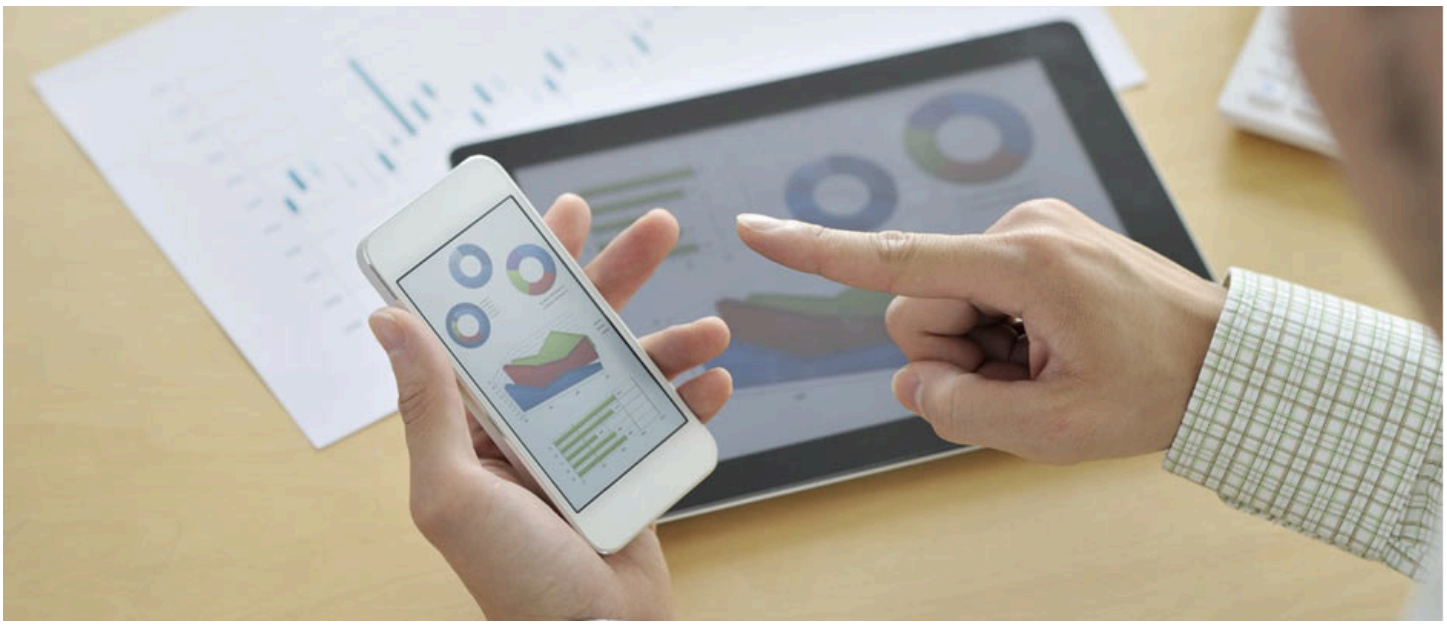
scientiamobile

*"The first step in a great mobile experience"*



# WURFL INFUZE MODULE FOR IIS: USER GUIDE

## Support

The ScientiaMobile Enterprise Support Portal is open to all WURFL users, both commercial license holders and evaluation users. It represents the combined knowledge base for the WURFL community. Commercial licensees are invited to post questions in the forum using the account to which their licenses are associated. This may mean faster handling of those posts by ScientiaMobile's personnel.

For commercial license holders, there are tiered support levels to address a variety of business support needs. After logging into your account, commercial licensees with support options can access the Enterprise Support portal to post tickets. These tickets will receive expedited attention.

To inquire about support plans, use our License Inquiry or our General Inquiry form.

## Update Notifications

If you would like to be notified of our API updates, major data updates, and other technical changes, please subscribe to our ScientiaMobile Announcements list.

scientiamobile

www.scientiamobile.com

Tel +1.703.310.6650

E-mail: sales@scientiamobile.com

Copyright © 2026 ScientiaMobile, all rights reserved. WURFL Cloud, WURFL OnSite, WURFL and, InFuze WURFL InSight and respective logos are trademarks of ScientiaMobile. Apache is the trademark of the Apache Software Foundation. NGINX is the trademark of Nginx Software Inc. Varnish is the trademark of Varnish Software AB

# WURFL InFuze Module for IIS: User Guide

This document will help you install and configure WURFL IIS on Windows Server 2003, 2008, and 2012. WURFL IIS is an IIS ISAPI plugin that adds selected HTTP headers to every request received by IIS Server. Added headers will contain values for the WURFL Static/Virtual Capabilities selected in the configuration file related to the User Agent of the current request.

# Installing the WURFL InFuze Module for IIS

WURFL IIS is distributed as a Windows installer package for 64 bit Windows versions.

Once the package is installed, go to your Server Manager -> IIS Manager, select the site you want to add WURFL IIS to -> ISAPI FIlter -> Add : Filter Name : WURFL IIS, Executable : `C:\Program Files\Scientiamobile\WURFL ISAPI\wurfl-isapi.dll` ***Warning:*** *Since version 1.8.3.0: the installation of* `libwurfl for Windows 1.8.3.0 or greater` *is required and the default installation folder is* `C:\Program Files\Scientiamobile\WURFLIsapi\` `libwurfl` *is provided in the Downloads section of your license in the Customer Vault*

# WURFL Data Snapshot

To perform lookups, you will need a copy of your WURFL data snapshot (also referred to as the `wurfl.xml`). While there is one included in the release package, it is intended to be a sample and will not contain all of your licensed capabilities. Your licensed WURFL data snapshot can be accessed by **following these directions**.

# Configuration

Once the plugin is installed, navigate to your installation folder and edit the `wurfl-conf.xml` configuration file to inject the headers that your web application will need.

## Single site configuration example

Let's assume our IIS Server hosts a single website **www.mywebsite.com** The `wurfl-conf.xml` configuration file should contain a single `<WurflIIS>` section whose `site` attribute value is the regex matching the site `FQN`.

```
<WurflIISServer>

    <WurflIIS name="default" site="www\.mywebsite\.com">

        <WurflRoot>wurfl.zip</WurflRoot>

        <!-- WURFL Updater allows seamless update of WURFL engine with new data
downloaded from Scientiamobile. -->
        <!-- Updater configuration must be done after <WurflRoot>, which must be a
writable path, and a wurfl.zip file must already be present in order for the updater
to check the file.-->
        <!-- Put your personal updater url taken from your Snapshot generator page in
the Customer Vault. -->
        <!-- WURFL Updater url file type and WurflRoot must match (.zip or .xml.gz )
-->
        <!-- Valid values for the updater checking frequency (how often the updater
checks for any new WURFL data file -->
        <!-- to be downloaded and used by the engine) are DAILY,WEEKLY  (default
value is DAILY) -->
        <!-- Updater log file (wurfl-updater-log.txt) may be found in WURFL Isapi
installation folder. -->
        <!-- The folder and WURFL data file must writable by IIS -->
        <!-- <WurflUpdater
frequency='DAILY'>https://data.scientiamobile.com/xxxxx/wurfl.zip</WurflUpdater> -->

        <!-- <WurflPatch>patch.xml</WurflPatch> -->
        <!-- <WurflPatch>...</WurflPatch> -->
        <!-- <WurflPatch>...</WurflPatch> -->

        <!-- WURFL Cache: one of the following-->
        <WurflCache params="100000">single_lru</WurflCache>
        <!-- <WurflCache>null</WurflCache> -->

        <!-- By default wurfl static/virtual capability names are mapped to
WURFLCAP_[upper(capability name)] http header
            hence for example static capability ux_full_desktop will be mapped to
HTTP header
            WURFLCAP_UX_FULL_DESKTOP -->
        <WurflRequestCapability>ux_full_desktop</WurflRequestCapability>
        <WurflRequestCapability>is_tablet</WurflRequestCapability>
        <WurflRequestCapability>is_wireless_device</WurflRequestCapability>

        <!-- You may have a custom HTTP header name specified in the tag attribute
"header" -->
        <WurflRequestCapability
header="DEVICE_OPERATING_SYSTEM">device_os</WurflRequestCapability>

        <!--
        <WurflRequestCapability>device_os_version</WurflRequestCapability>
```

```
        <WurflRequestCapability>pointing_method</WurflRequestCapability>
        <WurflRequestCapability>preferred_markup</WurflRequestCapability>
        <WurflRequestCapability>resolution_height</WurflRequestCapability>
        <WurflRequestCapability>resolution_width</WurflRequestCapability>
        <WurflRequestCapability>xhtml_support_level</WurflRequestCapability>
        <WurflRequestCapability>is_smarttv</WurflRequestCapability>
        <WurflRequestCapability>can_assign_phone_number</WurflRequestCapability>
        <WurflRequestCapability>brand_name</WurflRequestCapability>
        <WurflRequestCapability>model_name</WurflRequestCapability>
        <WurflRequestCapability>marketing_name</WurflRequestCapability>
        <WurflRequestCapability>is_touchscreen</WurflRequestCapability>
        -->

        <!-- Urls blacklist: all urls matching the following regex will be excluded
from wurfl headers injection -->
        <!-- <WurflDoNotProcessUrl> -->

            <!-- Excludes all .ico files from processing-->
            <!-- <WurflUrl name="BlackList-Rule-1">.*\.ico</WurflUrl> -->

            <!-- Excludes all files in "img" folder from processing-->
            <!-- <WurflUrl name="BlackList-Rule-2">.*\/img\/.*</WurflUrl> -->

        <!-- </WurflDoNotProcessUrl> -->

        <!-- Urls whitelist: all urls matching the following regex will be injected
with wurfl headers -->
                <!-- if no WurflProcessUrl section is present default action is to
inject, no need for a catchall rule -->
        <!-- <WurflProcessUrl> -->

            <!-- Includes all .jpg files in processing-->
            <!-- <WurflUrl name="WhiteList-Rule-2">.*\.jpg</WurflUrl> -->

        <!-- </WurflProcessUrl> -->

        <!-- <WurflLogHeaderInjection>true</WurflLogHeaderInjection> -->
        <!-- default -->
        <!-- <WurflLogHeaderInjection>false</WurflLogHeaderInjection> -->

    </WurflIIS>

</WurflIISServer>
```

You can insert as many `<WurflRequestCapability>` tags as you need in your `wurfl-conf.xml` file, with each static capability (or virtual capability) resulting in a new HTTP header. A full list of capabilities can be found **here**.

## Multiple sites configuration examples

Let's assume our IIS Server hosts two websites **www.firstwebsite.com** and **www.secondwebsite.com**. If we want to use different configurations for each website, the `wurfl-conf.xml` configuration file should contain two `<WurflIIS>` sections:

- one with `site` property value = `www\.firstwebsite\.com`
- one with `site` property value = `www\.secondwebsite\.com`

```
<WurflIISServer>

    <WurflIIS name="first" site="www\.firstwebsite\.com">
        <WurflRoot>wurfl.zip</WurflRoot>
        .
        .
    </WurflIIS>

    <WurflIIS name="second" site="www\.secondwebsite\.com">
        <WurflRoot>wurfl.zip</WurflRoot>
        .
        .
    </WurflIIS>
</WurflIISServer>
```

Let's assume our IIS Server hosts websites under `com` TLD and `net` TLD. If we want to use different configurations for each TLD, the `wurfl-conf.xml` configuration file should contain two `<WurflIIS>` sections:

- one with `site` property value = `.*\.com`
- one with `site` property value = `.*\.net`

```
<WurflIISServer>

    <WurflIIS name="com" site=".*\.com">
        <WurflRoot>wurfl.zip</WurflRoot>
        .
        .
    </WurflIIS>

    <WurflIIS name="net" site=".*\.net">
        <WurflRoot>wurfl.zip</WurflRoot>
        .
        .
    </WurflIIS>
</WurflIISServer>
```

If we want to use the same configurations for each TLD, the `wurfl-conf.xml` configuration file should contain only one `<WurflIIS>` section:

```
<WurflIISServer>

    <WurflIIS name="allsites" site=".*">
        <WurflRoot>wurfl.zip</WurflRoot>
        .
        .
    </WurflIIS>
</WurflIISServer>
```

# Configuration Tags details

| Tag | Description | Availability |
|---|---|---|
| **<WurflIISServer>** | Aggregates the WURFL engines configurations for local IIS Server<br>**Context:** `root`<br>**Type:** `mandatory` | 1.7.0.1 |
| **<WurflIIS>** | Contains the configuration for a subset (WURFL engine) of the web sites hosted by local IIS Server<br>**Type:** `mandatory`<br>**Context:** `<WurflIISServer>`<br>**Attributes:**<br>`name`: a symbolic name for the web sites managed by the nested configuration for logging purposes<br>`site`: a regex representing web site URLs managed by the nested configuration. WURFL Isapi filter will check the HTTP Request against this regex to determine which WURFL engine will manage the incoming HTTP Request. | 1.7.0.1 |
| **<WurflRoot>** | Defines the location (path) of the WURFL data file.<br>**Type:** `mandatory`<br>**Context:** `<WurflIIS>`<br>**Value:**: the full path of the WURFL data file | 1.6.4 |

| Tag | Description | Availability |
|---|---|---|
| **<WurflUpdater frequency="DAILY">** | Allows seamless update of the WURFL Engine with new data downloaded from Scientiamobile.<br>The tag `<WurflRoot>` must precede it.<br>**Type:** `not mandatory`<br>**Context:** `<WurflIIS>`<br>**Value:**: the `data url` (taken from the **Snapshot generator** page of your ScientiaMobile Customer Vault, choosing between two data file types: `.zip` or `.xml.gz`)<br>Take care that `<WurflRoot>` file type and `<WurflUpdater>` `data url` file type match so you may need to change the `<WurflRoot>` file type accordingly.<br>**Attributes:** `frequency` (how often the updater checks for any new WURFL data file to be downloaded and used by the engine) whose value you can choose between `DAILY` and `WEEKLY` (default value is `DAILY`)<br><br>In order to let the Updater perform its activities both the `<WurflRoot>` folder and file must be writable by IIS.<br>The `wurfl-updater-log.txt` file in WURFL Isapi installation folder will contains details on Updater activity. | 1.8.3 |
| **<WurflPatch>** | Adds one or more custom patch files to the WURFL repository.<br>**Type:** `not mandatory`<br>**Context:** `<WurflIIS>`<br>**Value:**: the full path of patch file | 1.6.4 |
| **<WurflEngineTarget>** | `This configuration option is deprecated and will be removed in a future release.` | 1.6.4 |
| **<WurflCache params="100000">single_lru<WurflCache>** or **<WurflCache>null<WurflCache>** | In order to increase performance while processing real HTTP traffic, we suggest setting up an `LRU` cache. The `LRU` caching strategy will speed up lookup operations on processed User Agents by keeping them in an LRU map. By default the cache will be set to 30000 entries, which accounts for 7 to 10 MB of additional memory usage. Specific concerns regarding memory usage apart, users are advised to size their cache generously (100,000 or more) to increase performance. For more information, please see LRU Cache Mechanism.<br>**Type:** `not mandatory`<br>**Context:** `<WurflIIS>` | 1.6.4 |
| **<WurflRequestCapability>** | Defines one or more WURFL Static/Virtual Capabilities values to be injected in the Request as HTTP headers.<br>**Type:** `not mandatory`<br>**Context:** `<WurflIIS>` | 1.6.4 |
| **<WurflUserAgentPriority>** | `This configuration option is deprecated and will be removed in a future release.` | 1.6.4 |
| **<WurflLogHeaderInjection>** | Logs header injection activity<br>**Type:** `not mandatory`<br>**Context:** `<WurflIIS>`<br>**Value:** `true`/`false` (default value is `false`) | 1.6.4 |

| Tag | Description | Availability |
| --- | --- | --- |
| **<WurflDoNotProcessUrl>** | URLs blacklist section: all URLs matching the contained regex will be excluded from wurfl headers injection.<br>**Context:** `<WurflIIS>` | 1.7.0.1 |
| **<WurflProcessUrl>** | URLs whitelist section: all URLs matching the contained regex will be injected with wurfl headers.<br>**Context:** `<WurflIIS>` | 1.7.0.1 |
| **<WurflUrl>** | An url to match<br>**Context:** `<WurflProcessUrl>` `<WurflDoNotProcessUrl>`<br>**Value:** the regex representing the url<br>**Attributes:** `name`: a symbolic name for the URL, for logging purposes | 1.7.0.1 |

---

**© 2026 ScientiaMobile Inc.**
**All Rights Reserved.**

**NOTICE:** All information contained herein is, and remains the property of ScientiaMobile Incorporated and its suppliers, if any. The intellectual and technical concepts contained herein are proprietary to ScientiaMobile Incorporated and its suppliers and may be covered by U.S. and Foreign Patents, patents in process, and are protected by trade secret or copyright law. Dissemination of this information or reproduction of this material is strictly forbidden unless prior written permission is obtained from ScientiaMobile Incorporated.