Sun Java System Web Server 6.1
SP6 FastCGI Plug-in Release
Notes
Release Notes for Sun Java™ System Web Server
6.1 FastCGI Plug-in

These release notes contain important information available at the time of release of Sun Java System Web Server 6.1 FastCGI plug-in, including information about server application functions (SAFs), installation, configuration, technical notes, and pointers to additional resources. Review the release notes prior to installing and configuring your software, and then periodically thereafter for the most up-to-date information.

In addition to the 32–bit platform support, Sun Java System Web Server 6.1 SP6 also provides support to 64–bit FastCGI.

The complete Sun Java System Web Server 6.1 documentation is available at http://docs.sun.com/app/docs/coll/1308.2

These release notes contain the following sections:

- “Introduction” on page 5
- “Server Application Functions (SAFs)” on page 6
- “Installing the FastCGI plug-in” on page 10
- “Configuring the FastCGI plug-in” on page 12
- “Known Problems” on page 14
- “How to Report Problems and Provide Feedback” on page 14
- “Additional Sun Resources” on page 15

Introduction

FastCGI (http://www.fastcgi.com) is an open extension to CGI (Common Gateway Interface), which is a standard for interfacing external applications with web servers. Like CGI, FastCGI applications run in separate, isolated processes. Some of the FastCGI’s advantages are as follows:

- Performance - FastCGI processes are persistent; they are reused to handle multiple requests. This solves the CGI performance problem of creating new processes for each request.
- Support for distributed computing - FastCGI provides the ability to run applications remotely, which is useful for distributing load.
Process isolation - FastCGI application does not crash or corrupt the core server or other applications.

The FastCGI plug-in allows the Sun Java System Web Server to safely work with third-party dynamic content generation technologies such as PHP, Perl and Python in a scalable way.

The FastCGI plug-in implements the client side of the FastCGI protocol. This protocol allows producers of dynamic content (and content engines) to implement a common and scalable API that can be utilized by many web servers.

Server Application Functions (SAFs)

The FastCGI plug-in provides the following Server Application Functions (SAFs):

- "auth-fastcgi" on page 6
- "responder-fastcgi" on page 7
- "filter-fastcgi" on page 7
- "error-fastcgi" on page 7

The various parameters and error-reason strings for the FastCGI SAFs are described in the following sections:

- “FastCGI SAF Parameters” on page 8
- “FastCGI SAF Error Reason Strings” on page 9

auth-fastcgi

The auth-fastcgi SAF is used in a PathCheck directive. This SAF is used to specify the FastCGI server application to be used for authorization.

The auth-fastcgi PathCheck SAF sends the request to the specified Authorizer FastCGI application. If the application has returned a status code of 200, the authorization has succeeded and the Web Server proceeds with request execution. Otherwise, the response that was sent by the FastCGI application is sent back to the user-agent.

The auth-fastcgi SAF is optional and should only be used to forward the requests to an authorizer FastCGI application.

For a list of parameters accepted by auth-fastcgi SAF, see “FastCGI SAF Parameters” on page 8.

The following obj.conf code example demonstrates the use of auth-fastcgi:

PathCheck fn="auth-fastcgi" app-path="/foo/bin/php" bind-path="fooUDS"

On the first request to the auth-fastcgi, FastCGI plug-in creates the FastCGI authorizer application processes and sends the request for evaluation.
**responder-fastcgi**

The responder-fastcgi SAF forwards a request to a FastCGI application that acts as a Responder, for further processing. responder-fastcgi sends the request data to the listener onto which the app-path application processes are listening. Once the application responds to the request, service-fastcgi process is the response headers before sending them back to the user-agent.

For a list of parameters accepted by responder-fastcgi SAF, see “FastCGI SAF Parameters” on page 8.

The following obj.conf code example demonstrates the use of responder-fastcgi:

```conf
Service fn="responder-fastcgi" app-path="/foo/bin/perl" bind-path="fooPerl"
```

**filter-fastcgi**

The filter-fastcgi SAF forwards a request to a FastCGI application that acts as a Filter, for further processing. A filter application receives the information associated with an HTTP request plus additional data from a file stored on the server, and generates a filtered version of the data stream as the response.

filter-fastcgi sends the request data in FastCGI record format to the application specified by app-path. Once the application responds to the request, filter-fastcgi process is the response headers before sending them to the user-agent.

For a list of parameters accepted by filter-fastcgi SAF, see “FastCGI SAF Parameters” on page 8.

The following obj.conf code example demonstrates the use of filter-fastcgi:

```conf
Service type="magnus-internal/perl" fn="filter-fastcgi" app-path="/foo/bin/perl" bind-path="fooPerl"
```

**error-fastcgi**

The error-fastcgi SAF is called when the FastCGI plug-in encounters an error. This is used to display a specified page or URL.

For a list of parameters accepted by error-fastcgi SAF, see “FastCGI SAF Parameters” on page 8.

The following obj.conf snippet demonstrates the use of error-fastcgi:

```conf
Error fn="error-fastcgi" error-reason="Fastcgi Connection Error" error-url="http://www.foo.com/errorPage1.html"
```
FastCGI SAF Parameters

FastCGI plug-in SAFs auth-fastcgi and responder-fastcgi both accept the following parameters unless otherwise mentioned explicitly:

**Note** – Parameters chroot, user, group and nice are applicable to only UNIX platforms. On Windows platforms, these parameters are ignored.

- **app-path** - (Optional) FastCGI server application path that processes the request. The functionality is dependent on the value of the “bind-path” as follows:
  - If only app-path is specified, the plug-in will create FastCGI server application. This server application will listen to UNIX Domain Sockets created by the plug-in.
  - If both app-path and bind-path are specified, the plug-in will create FastCGI server applications. This server application will listen to the TCP/IP address and port value specified in bind-path or UNIX Domain Sockets created by the FastCGI plug-in.
  - If only bind-path is specified, the plug-in will directly communicate with the FastCGI server application.
  - If both app-path and bind-path are not specified, then the plug-in will log an error message.

- **app-args** — (Optional) Values that are passed as the arguments to the FastCGI application process. Multiple app-args parameters are allowed. The format for the multiple app-args parameters is:

  app-args="value" app-args="value" ..

- **bind-path** - (Optional) Can be a UNIX Domain Socket (UDS) name or of the form host:port. The description of app-path parameter explains the usage of bind-path parameter. Note that the UDS name is applicable only on UNIX platforms; on Windows platforms, bind-path must be specified as host:port.

- **min-procs** - (Optional) Integer specifying the minimum number of FastCGI application processes to be created. Defaults to 1.

- **max-procs** - (Optional) Integer specifying the maximum number of FastCGI application processes that could be created at any time. It should be equal to or greater than min-procs. Defaults to 1.

- **chroot** - (Optional) Used to set the root directory of the chroot jail that the FastCGI server application runs in. Defaults to the server’s root directory.

- **user** - (Optional) Specifies the user ID the FastCGI server application runs as. Defaults to server’s user ID.

- **group** - (Optional) The FastCGI server application would be running under the specified group. Defaults to server’s group.

- **nice** - (Optional) Specifies the nice value of FastCGI server application processes.

- **listen-queue** - (Optional) Integer specifying the listen queue size for the socket. Defaults to 20.
- app-env - (Optional) Value pairs which will be passed as environment variables to the FastCGI server application process. Multiple app-env parameters are allowed. The format for multiple app-env parameters is:
  
  app-env="name=value" app-env="name=value" ....

- reuse-connection - (Optional) Boolean value that determines if connections to FastCGI server applications are reused. False (0, false, no) indicates that the connections to FastCGI server applications are closed after each request. True (1, true, yes) indicates that existing connections are reused for new requests. See also connection-timeout.

- connection-timeout - (Optional) If reuse-connection is true, this value specifies the timeout value in seconds for the pooled connections. If a connection is idle for the specified amount of time, the plug-in closes the connection. See also reuse-connection.

- resp-timeout - (Optional) Integer representing the FastCGI server response timeout in seconds. If there is no response from the FastCGI server application within the specified period of time, the request is discarded.

- restart-interval - (Optional) Integer representing the time interval (in minutes) after which the FastCGI server application is restarted. Defaults to 60 minutes (1 hour).

- req-retry - (Optional) Integer representing the number of times the plug-in should resend the request when the FastCGI server application rejects the request.

The error-fastcgi SAF accepts the following parameters:

- error-url - Specifies the page, URI or URL to be displayed in case a failure or error occurs. The value of this parameter can be an absolute path, a path relative to docroot, or an URL or URI.

- error-reason - (Optional) String representing the FastCGI protocol error. This is used to differentiate error URLs to be displayed, in case of any plug-in errors.

### FastCGI SAF Error Reason Strings

Here is the list of valid error-reason strings:

- Missing Parameters: whenever app-path and bind-path are not specified.
- “Stub Start Error”: unable to start the stub for various reasons like fork failure, bind error, listen error.
- “Stub Connection Failure”: client socket creation or connect failure.
- “No Permission”: stub executable has no exec permission, stat failure, and so on.
- “Stub Request Handling Error”: unable to send the request to stub, received invalid/no response from the stub for a request, and so on.
- “Set Parameter Failure”: when set user, group, chroot, nice (and so on) fail.
- “Invalid user and/or group”: when user and/or group is invalid.
- “Invalid Parameters”: when invalid nice value is specified.
- “Server Process Creation Failure”: application fork failure, application exec failure, bind error, listen error, and so on.
“Fastcgi Protocol Error”: invalid record version, invalid record type, unknown role, and so on.
“Internal Error”: unable to open the file to be sent to the filter application, any other unknown errors.

Installing the FastCGI plug-in

The FastCGI plug-in is available for use with the Sun Java System Web Server 6.1 or later Service Packs.

This section includes the following topics:
- "Package Contents" on page 10
- "Installing on Solaris, Linux, HP-UX and AIX Using tar Packaging" on page 11
- "Installing on Windows" on page 11
- "Installing on Solaris Using SVr4 Packaging" on page 11
- "Installing on Linux using RPM Packaging" on page 12

Package Contents

The contents of the platform specific packages are:

Solaris, Linux, AIX:
- plug-ins/fastcgi/README
- plug-ins/fastcgi/libfastcgi.so
- plug-ins/fastcgi/Fastcgistub

HP-UX:
- plug-ins/fastcgi/README
- plug-ins/fastcgi/libfastcgi.sl
- plug-ins/fastcgi/Fastcgistub

Windows:
- plug-ins\fastcgi\README
- plug-ins\fastcgi\fastcgi.dll
- plug-ins\fastcgi\Fastcgistub.exe

The FastCGI plug-in is packaged as a tarball, a zip file, an SVR4 package, or an RPM as package.
Installing the FastCGI plug-in

Installing on Solaris, Linux, HP-UX and AIX Using tar Packaging

$ gzip -d sun-webserver61-fastcgi-{sol|lin|hpux|aix}.tar.gz

;; Uncompress the tar archive,
;; where {sol|lin|hpux|aix} reflects
;; the operating system
;; environment the library will
;; be used

$ tar xvf sun-webserver61-fastcgi-{sol|lin|hpux|aix}.tar

;; Extract the tar archive

Installing on Windows

$ unzip sun-webserver61-fastcgi-win.zip

;; Uncompress the ZIP archive

Installing on Solaris Using SVr4 Packaging

$ su

;; root access is required to install
;; SVr4 packages

$ cd <path/to/package>

;; Change directory to where the package
;; is located

# pkgadd -d .

;; Install SUNWwbsvr-fastcgi.pkg package

Note – This installation places the shared object and README in
/opt/SUNWwbsvr/plug-ins/fastcgi.
Installing on Linux using RPM Packaging

```bash
$ su

;; root access is required to
;; install RPM packages

$ cd <path/to/package>

;; Change directory to where
;; the package is located

# rpm -iUvh sun-webserver-fastcgi.rpm

;; Install
;; sun-webserver-fastcgi.rpm
;; package
```

Note – This installation places the shared object and README in /opt/sun/webserver/plug-ins/fastcgi.

Please refer to "Configuring the FastCGI plug-in."

Configuring the FastCGI plug-in

The FastCGI plug-in is configured through the existing magnus.conf and obj.conf Web Server configuration files as documented in the following subsections.

- “magnus.conf” on page 12
- “obj.conf” on page 13

magnus.conf

The existing load-modules SAF is used to load the FastCGI plug-in shared object. The shared object filename varies by platform as follows:

- libfastcgi.so (Solaris, Linux, AIX)
- libfastcgi.sl (HP-UX)
- fastcgi.dll (Windows)

The following magnus.conf code example demonstrates the use of load-modules to load the FastCGI plug-in:

```ini
Init fn="load-modules" shlib="/opt/SUNWwbsvr/plug-ins/fastcgi/libfastcgi.so"
```
The FastCGI plug-in introduces the following SAFs that can be referenced in obj.conf:

- auth-fastcgi
- responder-fastcgi
- filter-fastcgi
- error-fastcgi

For example, the following obj.conf code example configures the server to forward requests in the /fcgi/php and /fcgi/perl URI space to fcgi.php and fcgi.perl respectively, this in turn is configured to pass the requests to FastCGI module for further processing.

```xml
<Object name="default">
  NameTrans fn="assign-name"
  from="/fcgi/php(\/*)"
  name="fcgi.php"
  NameTrans fn="assign-name"
  from="/fcgi/perl(\/*)"
  name="fcgi.perl"
  ...
  Service type="magnus-internal/perl" fn="filter-fastcgi"
  \<other parameters required for running Filter FastCGI processes>\n  \...
  \</Object>\n
<Object name="fcgi.php">
  PathCheck fn="auth-fastcgi"
  \<other parameters required for running Authorizer FastCGI processes>\n  Service fn="responder-fastcgi"
  \<other parameters required for running Responder FastCGI processes>\n  \</Object>\n
<Object name="fcgi.perl">
  PathCheck fn="auth-fastcgi"
  \<other parameters required for running Authorizer FastCGI processes>\n  Service type="magnus-internal/perl" fn="responder-fastcgi"
  \<other parameters required for running Responder FastCGI processes>\n  Error fn="error-fastcgi" error.reason="Fastcgi Connection Error"
  error-url="http://www.foo.com/errorPage1.html"
  \</Object>\n```

**Note** – FastCGI SAFs can be invoked in different ways as mentioned above; that is, defining different Objects for different URL patterns or mapping them to different MIME types.

- “Known Problems” on page 14
Known Problems

This section includes the known problems with the Sun Java System Web Server 6.1 FastCGI plug-in.

TABLE 1 Known Problems in Sun Java System Web Server 6.1 FastCGI plug-in

<table>
<thead>
<tr>
<th>Problem ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6228687</td>
<td>(On Linux 2.1 Advanced Server only) caution - Fastcgistub process is restarted when the parent webservd process is killed.</td>
</tr>
</tbody>
</table>

How to Report Problems and Provide Feedback

If you have problems with Sun Java System Web Server 6.1 FastCGI plug-in, contact Sun customer support using one of the following mechanisms:

- Sun Software Support services online at [http://www.sun.com/service/support/software/](http://www.sun.com/service/support/software/)
- The telephone dispatch number associated with your maintenance contract

So that we can best assist you in resolving problems, please have the following information available when you contact support:

- Description of the problem, including the situation where the problem occurs and its impact on your operation
- Machine type, operating system version, and product version, including any patches and other software that might be affecting the problem
- Detailed steps on the methods you have used to reproduce the problem
- Any error logs or core dumps

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. Send your comments to Sun using the "Send comments" link at: [http://docs.sun.com/](http://docs.sun.com/)

Please include identifying information with your comments, such as the book’s part number and title.
Additional Sun Resources

Useful Sun Java System information can be found at the following locations:

- Documentation for Sun Java System Web Server 6.1 and Service Packs
  http://docs.sun.com/app/docs/coll/1308.2
- Sun Java System Software Products and Service
  http://www.sun.com/software
- Sun Java System Developer Information
  http://developers.sun.com/
- Sun Developer Support Services
  http://developers.sun.com/services/index.jsp
- Sun Java System Software Support Services
  http://www.sun.com/service/support/software/
- Sun Support and Training Services
  http://www.sun.com/training/
- Sun Java System Consulting and Professional Services
  http://www.sun.com/service/sunjavasystem/sjsservicessuite.html