

FSI *SERVER* 2.0

The Complete Single Source Imaging Solution

User Manual



FSI SERVER

Developed by:

NeptuneLabs GmbH
Lagesche Str. 32
D-32657 Lemgo
Germany

© 2009-2011 NeptuneLabs. All rights reserved.

Last updated: July 2011

FSI Server Version	2.0
Manual Revision	010

All brands and product names are trademarks or registered trademarks of the respective producers.

FSI Viewer is a registered trademark of NeptuneLabs GmbH, Germany.

Table of Contents

Introduction	8
Part I • Administrator Documentation.....	9
1. System Requirements	9
2. Installation.....	9
2.1 Installation on Microsoft Windows.....	9
2.2 Platform independent installation.....	10
2.3 Indexing Services	10
2.3.1 Windows	10
2.3.2 Mac OS X.....	10
2.3.3 Linux.....	11
2.3.4 Other common indexing services.....	11
3. Configuration	11
3.1 Configuration Tasks and Preconsiderations	11
3.2 Users, Groups, Permission Sets and Import Profiles.....	12
3.2.1 Permission Sets.....	12
3.2.2 Groups	13
3.2.3 Import Profiles	13
3.2.4 Examples.....	13
3.3 User Preferences	14
3.4 Configuration using FSI Administrator	15
3.4.1 Common Tasks.....	15
3.4.2 Advanced Configuration	16
3.5 Manual Configuration	18
3.5.1 Location of the configuration files	18
3.5.2 Importer component setting.....	18
3.5.3 Importer Profiles	22
3.5.4 Permission Sets.....	23
3.5.5 Groups	23
3.5.6 Users	24
3.5.7 Realtime image server component settings	24
3.5.8 Server profiles.....	25

4.	Licensing	27
5.	Logging.....	27
6.	Templates.....	27
 Part II • User Documentation		 28
1.	Getting Started.....	28
2.	Interface Overview	28
3.	Logging In	28
3.1	Search Bar	30
3.2	View Bar	30
3.3	Log Out	30
3.4	Folder Tree Menu	30
3.5	Location Bar.....	30
3.6	Folder Tree	31
3.7	Thumbnail Canvas	31
3.8	Tool Bar	31
3.9	Status Bar.....	31
3.10	Image Details Popup.....	32
4.	Managing Images.....	32
4.1	Uploading	32
4.2	Downloading	33
4.3	Deleting.....	33
4.4	Moving and Renaming	33
4.5	Searching	33
4.5.1	Full Text keyword search.....	33
4.5.2	Comparative search.....	34
4.5.3	Search modifiers	34
5.	Publishing Single Images.....	34
5.1	HTML Image	35
5.2	FSI Viewer	35
6.	Publishing Image Collections.....	38
6.1	FSI Viewer 360.....	38

6.2	FSI Pages and FSI Pages mobile	38
6.3	FSI Showcase.....	39
7.	Publishing Stereoscopic images.....	39
7.1	Publishing a single stereoscopic image	39
7.2	Publishing a stereoscopic image in FSI Viewer	40
7.3	Publishing stereoscopic images as FSI Viewer 360	40
8.	Server Addressing	40
8.1	Source.....	40
8.2	Type.....	40
8.3	Width and Height	41
8.4	Rect	41
8.5	Left, Top, Right, Bottom.....	41
8.6	Quality.....	41
8.7	Tpl	41
8.8	Effects	41
8.9	Profile.....	42
9.	Real-time Image Effects	43
	Appendix	53
A	Supported Input/Output Formats.....	53
1.	Input Formats	53
2.	Output Formats.....	53
B	Server Addressing Quick Reference	54
1.	Image Requests	54
2.	Info Requests.....	54
3.	List Requests.....	55
4.	Search Requests.....	55
C	Tomcat and JVM Settings.....	56
1.	Java Virtual Machine Settings	56
2.	AJP & JK.....	56
D	Search field reference	57

1.	Textual search field.....	57
2.	Comparative search fields	58
E	Updating from FSI Server 1 to 2	58
1.	Update using the Windows installer version.....	58
2.	Updating the platform independent version	58
3.	User configuration	59
4.	Import Scanner	59
5.	Search database configuration.....	59

NeptuneLabs FSI Server

Next Generation Single Source Imaging

Introduction

FSI Viewer Server is a high performance Java Imaging Server, designed for outstanding scalability (multi core CPUs, multi server setups, load balancing) and excels in superior image quality compared to all imaging servers supported by FSI Viewer in the past.

While supporting Single Source Imaging (e.g. JPEG and PNG images for web sites) it also has full FSI Viewer support including FSI Showcase and FSI Pages.

The HTML/AJAX based web interface makes uploading, managing and publishing images as easy as handling local files. It also offers quick access to more than 20 real-time image effects and all the FSI Viewer publishing options.

Key features

- internal storage containing image copies optimized for fast delivery
- support for transparent images
- fast, high quality real-time image scaler
- support for large images
- more than 20 combinable real time image effects
- support for read only source image pools
- input format independent performance

Thank you for using NeptuneLabs software!

www.fsi-viewer.com - Online Resources for FSI Viewer & FSI Server

Visit www.fsi-viewer.com for software updates, regularly updated samples, tutorials and downloads.

NeptuneLabs FSI Server

Part I • Administrator Documentation

1. System Requirements

FSI Server – Minimum Hardware Requirements

- Dual Core CPU or better recommended
- 2 GB RAM per CPU core recommended

FSI Server – Software Requirements

- Java SE 6 (Oracle/Sun Java SE 6)
- Java Application server complying with servlet specification 2.5 (Tomcat)
- 32 Bit operating system (64 Bit operating system highly recommended)

FSI Server has been successfully tested with the following operating systems:

- Linux (x86_64): Debian 4.0R9, 5.0.8, 6.0.1a
Fedora 9, 11, 14, CentOS 5.1, 5.5
Ubuntu 6.06.2 LTS, 8.04.3 LTS, 10.04.1 LTS
SuSE 11.1, 11.4
- Windows Server 2003, 2008
- Mac OS X 10.5, 10.6, Mac OS X Server 10.6
- OpenSolaris 2009.06

FSI Server webinterface requires one of the following browsers

- Internet Explorer 6 or newer
- Firefox 3 or newer
- Safari 3 or newer
- Google Chrome

2. Installation

The current version of FSI Server is available as an installer for Microsoft Windows or as a Java WebArchive (war) for deployment in an existing application server.

2.1 Installation on Microsoft Windows

The installation for Windows is a packed executable file for either 32 or 64 Bit Windows systems. The installer contains the current Sun Java Runtime

Environment and the Apache Tomcat servlet container, therefore no further software is required prior to installation. To start the installation double click the downloaded installer. Choose the installation directory and follow the instructions on the screen.

2.2 Platform independent installation

To install the FSI Server WebArchive a servlet container complying with the Java Servlet Specification 2.5 and running von a Java 6 Runtime Environment is required.

FSI Server has been developed and tested on Apache Tomcat 6 and Oracle/Sun Microsystems Java SE 6. FSI Server is installed by simply deploying the downloaded war-file. Please also see Appendix C for required and recommended Tomcat and Java Virtual Machine Settings.

2.3 Indexing Services

The internal storage will contain lots of files and directories. As this data is not relevant for any other application, it should not be scanned by any indexing services. Allowing such a service to scan the internal storage will lead to a noticeable load and therefore to a reduced server performance whenever the indexing service runs.

2.3.1 Windows

If the FSI Server was installed using the Windows installer version, the storage folder will be set up so that it is not indexed. If the FSI Server was installed using the platform independent version or the storage was chosen to be in an other folder by changing the configuration, then the storage folder should be excluded from indexing by right clicking the folder in the Windows explorer, choosing the advanced properties and clearing the checkbox from the indexing option.

2.3.2 Mac OS X

On Mac OS X the storage folder can be excluded from the spotlight index by opening the spotlight preferences from the spotlight dropdown menu and selecting the 'privacy' tab. The little "+" button will allow adding a location that will then no longer be indexed.

2.3.3 Linux

Most Linux distributions come with support for the locate command as part of the find-utils. The updatedb process can be configured to exclude paths. Depending on the distribution the configuration is most likely to be found in `/etc/updatedb.conf` or `/etc/sysconfig/locate`.

2.3.4 Other common indexing services

There are some other commonly used indexing services, often referred to as desktop searches. Amongst others, these are the Google Desktop, Beagle or the Copernic Desktop Search. These should also all be deactivated or be configured to exclude FSI Servers internal storage.

3. Configuration

FSI Server consists of two core components: the importer component that imports images into the internal storage and the actual server component that delivers images and metadata to the web. The two components have separate configurations.

FSI Server can be configured by manually editing the XML configuration files or through a web-based graphical interface using FSI Administrator.

3.1 Configuration Tasks and Preconsiderations

Every FSI Server installation provides the same set of features up to the point where the licence key is installed. An unlicensed FSI Server installation provides the full set of features to allow interested customers to evaluate the full potential of the software. Depending on the licence key installed some features are disabled. Especially the user and group management differs in ASP and non-ASP Editions.

Basically FSI Server is delivered with a configuration suitable for most use cases. The default configuration was carefully chosen to minimize the necessary adaptations. To start using FSI Server, all that needs to be done is install the licence key, create a user account and define an import profile, which includes the path to your images.

As mentioned above, FSI Server imports the provided images into its internal storage. This import process converts and scales the source images which uses a considerable amount of the system resources and, depending on the amount and size of the source images, may take quite a long time. Due to the internal storage architecture the images in an import profile need to be re-imported if the profile is renamed. Choosing the correct name right from the beginning is therefore advisable.

3.2 Users, Groups, Permission Sets and Import Profiles

FSI Server uses group based rules to define the access permissions to certain files and directories. Every user belongs to one or more groups and every group has certain permissions restricting its members access to an import profile. The permissions are summarized in a "permission set" and every group import profile assignment references a permission set.

In the ASP edition FSI Server has three predefined access levels to further distinguish the users role in the system.

Normal Users: Normal users can access image data according to the group based permissions described above.

Administrators: Administrators have full access to all importer profiles and images.

Service Users: Service accounts do not have any permissions to access images. These accounts can be used to retrieve the server status information without allowing access to any images data.

In the Non-ASP editions of FSI Server *Administrator* and *Normal User* accounts are treated equally. Both have access to the server status and the file access permissions for both are defined by the permission sets.

3.2.1 Permission Sets

A permission set is a named set of authorizations. In detail these authorizations are:

imageread: permission to access images
inforead: permission to access image metadata
imagelist: permission to list images in a folder
dirlist: permission to list subfolders in a folder
upload: permission to upload files and create folders
delete: permission to delete files and folders

Please note: replacing or renaming an image requires upload as well as delete permission. There are three default permission sets called "public", "r" (read) and "rw" (read/write) which should cover most use cases:

	access images	access metadata	list files	list folders	upload	delete
public	•	•	•	—	—	—
r	•	•	•	•	—	—
rw	•	•	•	•	•	•

3.2.2 Groups

There are two default groups called "users" and "public". These are considered special groups and like the default permission sets they cannot be modified or deleted. All users are by default in the "users" group, this for example is useful to share a sample image folder. The "public" group is used to specify the access permissions for anonymous users (normal website visitors).

Non-ASP editions of FSI Server do not allow creating custom groups, therefore all permissions are handled by the "users" group so all registered users work on the same set of images.

3.2.3 Import Profiles

An import profile defines a directory and a set of rules on how to import this directory into FSI Servers internal storage. All images in this directory and its subfolders will be imported and (depending on the access permission) are then available via FSI Server under the import profiles name. The import profile declaration defines which group or groups have access to the directory contents by assigning a permission set.

3.2.4 Examples

To clarify the relationship between users, groups and permission sets, here is an example configuration. The import profile "Samples" has two group relations:

group "users" with permission set "rw":

Every user in the group "users" has full access to the content and can view, list, upload and delete contents.

group "public" with permissionset "public":

Every anonymous user can access image data, image metadata and list the images in a folder.

3.3 User Preferences

Preferences are environment variables that influence the behavior of the FSI Server web interface. Preferences can be set to apply to all users or to a single user account. Preferences can be set for all users or for a single user. If a preference is set for all users and also set for a single user, the value set for the single user will override the value set for all users. Once set the preferences will apply as soon as the user logs in to the webinterface.

In the current version of the webinterface delivered with FSI Server 2.0 the following preferences can be set to modify the code snip generation in the "Publish to web" interface:

viewerPath: Defines an alternate path to the FSI Viewer installation used. If this is set the FSI Viewer in this location will be used instead of the default viewer.

proxyURL: Defines the proxy or FSI Cache used to publish images.

proxyServerPattern: Redefines the path prefix in the proxy (FSI Cache profile name) used to access images and metadata.

proxyViewerPath: Redefines the subdirectory where the default FSI Viewer can be found on the proxy.

proxyJSPath: Redefines the subdirectory where the JavaScript helper files like fsiwriter can be found.

proxyUserViewerBase: Redefines the first part of the path to the user based FSI Viewer used to retrieve the viewer via the proxy.

If a preference is set for all users and also set for a single user, the value set for the single user will override the value set for all users. Once set the preferences will apply as soon as the user logs in to the webinterface.

3.4 Configuration using FSI Administrator

To use FSI Servers graphical configuration interface, you need a connected FSI Administrator. For details on installing FSI Administrator and connecting to an FSI application please refer to the FSI Administrator manual.

Once connected, you can enter the FSI Server configuration interface by clicking on the FSI Server logo in the "My Software" tab in FSI Administrator.

The every-day tasks are grouped in the "Common Tasks" menu. More complex tasks will be handled by wizards, which will lead through the task step by step. The more advanced and not so often used settings are accessible by choosing an entry from the "Advanced" menu.

3.4.1 Common Tasks

Often performed administration tasks have been grouped in the common tasks menu for quick access. Most of the interfaces for these tasks are self explaining but reading *Chapters 3.1 - 3.3* before starting is recommended.

New Account

The New Account Wizard will lead through the process of creating a new user account. In the ASP edition of FSI Server it will also allow creating a new group and importer profile. The first step is to choose the account type. The following screens allow choosing a username and password and, if appropriate, a group, import profile name and an input directory.

Preferences

The preferences management is split into default preferences that apply to all users and preferences that apply to a single user only. When editing the preferences for a single user the default preferences are shown as well. For details on preferences in general see *section 3.3*.

FSI Viewer

Managing the FSI Viewer installations is a common task. It involves setting up an extra FSI Viewer for a specific user or managing FSI Viewer skins. Installing and FSI Viewer for a single user will copy the default viewer to a user specific directory and set a user preference so that the new FSI Viewer location will be used in the publishing dialogs. Removing a user FSI Viewer will delete the users copy of FSI Viewer. This will break all already published codesnips and should therefore only be done by experienced users. Skins can be uploaded or deleted for all copies of the FSI Viewer, the default as well as the user viewers. In very rare occasions, when

a custom skin is used that involves more files than normal skins, manual deletion of files may be necessary. For backup purposes the FSI Viewer management interface allows downloading a zip of a complete FSI Viewer installation directory.

Licences

As of version 2.0 FSI Server supports managing licences for client applications like the FSI Viewer. All licences, the FSI Servers own licence as well as the client application licences can be managed in one interface. The interface will show a list of all installed licences and allows uploaded new licences as well as replacing or deleting licences.

3.4.2 Advanced Configuration

Administration tasks that need to be performed only once for the initial configuration and settings that are necessary for special use cases only are grouped in the "Advanced" menu.

Server Settings

The server settings interface allows editing the configuration of the real-time server component and provides access to the server profiles. A detailed description of the parameter values can be found in the interface help texts accessible by pressing the right mouse button or in *Chapters 3.5.7. and 3.5.8.*

Importer Settings

The importer settings interface allows editing the configuration of the importer component and provides access to the importer profiles. Please note that in the ASP editions an importer profile may also be created by the "New Account" wizard. For a detailed description of the available settings please refer to the interface help texts accessible by pressing the right mouse button or *see chapters 3.5.2. and 3.5.3.*

Users

This section allows managing user accounts. Creating accounts is possible but using the new account wizard mentioned above should be preferred.

Groups

In the ASP editions of FSI Server the groups interface allows creating and deleting groups. It also allows assigning users to groups or removing them from a group. For a description of groups in general please *see chapter 3.2.2.*

Permission Sets

As described in *Chapter 3.2.1* permission sets are used to define the operations a group member is allowed to perform on a certain import profile. This interface allows creating new permission sets from scratch, creating copies of existing permission sets as well as deleting permission sets. Permission sets currently in use can not be deleted.

Server Cache

FSI Server uses four internal memory caches to increase performance. The memory caches are not persistent and will be cleared on every FSI Server restart. This interface shows the four caches including the number of cached objects and allows deleting the cache without a server restart.

Search Database

The FSI Server uses a database to provide quick access to image metadata. This is primarily used for the search in the FSI Server webinterface. This screen shows detailed information on the database and the stored data sets. The database can be rebuilt from the original image metadata, but not that the search will deliver incomplete results during the rebuild process.

3.5 Manual Configuration

As mentioned in the introduction to this chapter FSI Server can be configured manually by editing the XML configuration files.

3.5.1 Location of the configuration files

The location of FSI Servers configuration files depends on the type of setup you choose. This directory is referred to as [CONFIG DIRECTORY] in the following chapters.

a) Platform independent installer

Using the platform independent installer the location is:

```
[webapps]/fsi/WEB-INF/
```

b) Windows installer

Using the Window installer the location is:

```
[SETUP DIRECTORY]\FSI Server\configuration\fsi-server\config\
```

FSI Server also supports using a custom configuration directory which allows easier updating when placed outside of the applications context. If the system property *com.neptunelabs.fsiserver.config.home* is set, its value is interpreted as an absolute path and will be used as the configuration directory. The system property can usually be set by using the -D option in the application servers startup parameters.

e.g. `demo.neptunelabs.fsiserver2.config=/etc/fsi/fsi-server`

3.5.2 Importer component setting

The importer component settings define the general behavior of the importer process which synchronizes the original source images with the internal storage. The importer settings are located in [CONFIG DIRECTORY]/config/importer.xml. This file is split into six main sections:

Application

The application section defines general settings for the importer component. By setting the enabled attribute to false the importer can be disabled completely.

Storage

Specifies the location of the internal storage. The given directory should exist and should be writeable by the user running the application server. If a relative path is given it will be interpreted as being relative to the web applications WEB-INF folder.

This setting should be identical to the storage setting in the server settings (see 3.5.7)

Imagethreads

The number of simultaneous image conversions. The value "auto" will determine the optimal setting depending on the amount of memory and the number of available CPU cores.

Maxcpu

The number of CPU threads to use for parallel internal tasks like scaling and encoding. "Auto" will determine the number of CPU threads available on the system.

Importstrategy

The import process can consume a lot of system resources. This parameter allows you to select the normal server use case, which influences how many threads will be used to import images and with which priority these threads will run. The actual number of used threads will depend on the amount of memory, the number of CPU cores and the "maxcpu" setting. Possible values are: BACKGROUND: All import tasks will be performed with minimum priority trying to have as less impact on other processes as possible. Use this when there are additional time critical applications other than FSI Server installed.

SHARED: This is the optimal value for most FSI Server setup and tries to share to the system resources between the importer and the server component.

STANDALONE: The standalone setting tries to optimize the importer behavior for servers that are dedicated to importing. This setting might cause a lot of i/o operations that might slow down other processes delivering content.

Prefetch

Enables loading image file data into memory before extracting the actual image data. In the current FSI Server version this only applies to Tiff and FPX input files. This uses more memory but will reduce IO-lock times. The default is false.

Prefetchmaxsize

The maximum file size allowed for prefetching image files. Larger files will be read from disk without prefetching.

Parallelread, parallelwrite, parallelreadwrite

These parameters allows controlling the servers internal file io controller mechanisms and should only be changed if you are absolutely sure what you are doing. The default settings are the optimal for most situations. Parallelread enables parallel file reading operations, multiple files can be read in parallel, the default is true. Parallelwrite allows writing multiple files at the same time. This results in higher load on the io subsystem as, in theory, the hard discs head needs to

perform more positioning operations compared to serialized file writing. The default settings for `parallelwrite` is `false`. `Parallelreadwrite` enables reading options to take place even if there are writing operations taking place. The default setting for this parameter is `false`.

Standalone

Allows or disallows an FSI Administrator instance to connect to this FSI Server. Note that setting `standalone` to `false` and not connecting an FSI Administrator may be a security risk. Please refer to the FSI Administrator manual for a detailed description.

Tempdirectory

Specifies the directory used to store temporary files. This will default to `"WEB-INF/work"` if nothing is specified.

Scanner

The scanner section controls the behavior of the scanning process, which monitors the input directories defined in the importer profiles. Setting the scanner sections `enabled` attribute to `false`, will disable the scanning process. New images placed in the filesystem will not be found and must therefore be uploaded using the webinterface or the OpenAPI.

Sorting

This specifies if the contents of a directory should be sorted before it is processed. Enabling sorting will help that new images in a folder will be imported in the correct order. Sorting the directory contents on every scan might have an impact on the server's performance, especially with large directories.

Convertbefore1stscan

If set to `true` the FSI Server will start importing images immediately. If not, then newly found images will be enqueued but not imported before the first full scan after a restart of the server has been completed.

Search

The search section allows specifying the settings for the search database. These should match the search section in the real-time components setting (*see 3.5.7*)

Type

Defines the type of database to be used. Currently supported databases are the H2 database from `h2database.com` as well as `MySQL/MariaDB`. By default an embedded version of the H2 database is used.

Storing the searchable metadata on an external database server requires purchasing an additional FSI Server Plug-in. Please contact your local reseller or NeptuneLabs directly for more information.

Hostname and port

Define the hostname and the port which will be used to connect to the database.

Database

Defines the database name used to store the search data.

Username and password

Specify the credentials used to log in to the database.

Options

Allows providing database specific options to use when connecting to the database.

ColorManagementSystem

The colormangement section defines the settings for the colormangement used when importing images. Disabling this section will disable the colormangement systems and no color profiles will be used when importing images.

```
<DefaultCMYKProfile>auto</DefaultCMYKProfile>  
<DefaultRGBProfile>auto</DefaultRGBProfile>  
<DefaultGrayProfile>auto</DefaultGrayProfile>  
<DefaultLabProfile>auto</DefaultLabProfile>
```

These settings define the default profiles to be used when importing images that do not contain color profiles. Auto will evaluate to an internal profile. Other valid values would be complete paths to valid color profile files.

Formats

The formats section contains a list for `<format>` nodes containing file types that should be imported. Other files will be ignored.

Protected dirs

The protected dirs section contains a list of `<protected_dir>` nodes containing names of directories that should be skipped while scanning for new files. The directory names can contain a `*` as wildcard.

Protected files

The protected files section contains a list of `<file>` nodes containing names of files that should be skipped while scanning.

Ping Service

This section contains a list of <address> nodes defining the remote ip addresses or hostnames allowed to access the Ping service.

3.5.3 Importer Profiles

The profiles are defined by files in the `→[CONFIG_DIRECTORY]/importer_profiles` directory of the FSI Server installation that end with the suffix `.xml`. A profile defines where to look for image files, how to import them and if they can be modified via the webinterface.

Enabled

The enabled node allows enabling and disabling a profile. A disabled profile is ignored and treated as nonexistent. Files that were imported via this profile will therefore be deleted from the internal storage when a profile is deleted.

Origin

The origin section defines where to look for images and how to access the found images.

Accessor

The accessor option defines how to access the profiles resources. Currently filesystem is the only supported accessor type.

Location

The location option defines where to look for images. A relative path is interpreted to be relative to the applications `WEB-INF/` directory.

ReadOnly

If the readonly option is enabled uploading or modifying the profiles directory content via the webinterface is prohibited.

Download

Defines if the original image file is downloadable via the webinterface for authenticated users.

Conversion

The conversion section defines parameters applied when importing files.

Format

Specifies the compression format of the image data in the internal storage. Valid compression formats are JPEG and LOSSLESS which uses a Deflate compression algorithm. This setting has a large impact on the hard disk size used by the internal storage.

Quality

The JPEG compression quality used internally only applies if the format is set to JPEG.

Access

This sections assigns groups and permissionsets to the import profile (see *section 3.2* for details). Every entry in this section must be a <group> node with the attributes name and permissionset, e.g.

```
<group name="users" permissionset="rw" />
```

3.5.4 Permission Sets

Permission sets are defined in [CONFIG DIRECTORY]/config/permissionsets.xml. Each permission set is defined by a <permissionset> node.

The name of the permission set is set using an attribute called "name". Every permission set node must have six childnodes named "imageread", "inforead", "imagelist", "dirlist", "upload" and "delete", referring to the permissions described in *3.2.1* Each of these nodes can have a value of "true" or "false". If the permissionsets.xml is not present on the system, then only the three default permission sets will be used.

3.5.5 Groups

The groups are defined in [CONFIG DIRECTORY]/config/groups.xml. Each group is represented by a node called <group>. The group name is specified by an attribute called "name". For every user belonging to a group, the group node needs a child node called user with the users login as text value. In non-asp editions the groups file is not read and only the two default groups as described in *3.2.2* are available.

3.5.6 Users

Users are defined in [CONFIG DIRECTORY]/config/users.xml. This file contains `<user>` node for each user. The username (login) is defined in the attribute "name". The password can be provided as plain text in an attribute called "plain" or as a SHA-256 hash in an attribute called "hash". A third attribute called "accesslevel" defines the users role in the system. The access level can be set to one of the three possible values according to the levels described *in 3.2*.

"normal": for standard users.
"admin": for administrators.
"service": for service users.

3.5.7 Realtime image server component settings

The server settings are defined in [CONFIG DIRECTORY]/config/settings.xml]. This file has three main sections called application, search and limits. These sections define general settings for the real-time server component and allow disabling the component completely.

Application

The parameters in the application section contain basic configuration options of the server component. The server component can be completely deactivated by setting the enabled attribute of the application tag to false.

Storage

The storage option specifies where the internal storage can be found. If located on the same machine, it should be identical to the storage setting in the importer settings. If a relative path is specified, it will be relative to the web applications `WEB-INF/` directory.

ServerTokens

Defines the response headers sent by FSI Server. Possible values are "full", "prod", "debug", "none".

Importer

This setting tells the real-time component where to find the importer. This is used to request the importer to prioritize a specific image that is currently in the import queue.

Copyrightnotice

This allows defining a customized string that will be embedded into the headers of the images delivered by the server. This allows identifying the images generated by FSI Server.

Search

The search section describes how to access the database containing the

search data gathered by the importer component. The settings in this section should match the settings in the search section of the importer configuration.

Limits

The limits section allows defining global default and max values that will be applied to incoming requests.

MaxWidth, MaxHeight

Define the maximum allowed values for width and height. These values can be overridden for specific server profiles by defining other values in the server profile configuration.

DefaultWidth, DefaultHeight

Define the default image dimensions that will be applied if there are no dimensions provided in the request parameters. These values can also be overridden on a per profile basis.

DefaultProfile

Defines the profile to be used if no profile parameter is given in the request.

3.5.8 Server profiles

Server profiles allow defining default values or restrictions that apply to every incoming request. Every request is processed using the rules in a server profile. If no profile name is given in the HTTP parameters, the default server profile will be used. A profile defines how to render an image or a meta data response, which default values the renderer will use and which file format is returned. The profile name is given by the filename. Each profile definition may include up to six sections:

Image Renderer (<imagerenderer>)

Defines the renderer to use when an image request is directed at this profile and which default values and limits apply when rendering the image. The renderer to use by selected by specifying an "implementation" node which contains the class name of the renderer. If the "implementation" node is omitted then the default renderer will be used.

Depending on the chosen renderer implementation further nodes may define parameters for the renderer.

Info Renderer (<inforenderer>)

Currently only the default info renderer is delivered with FSI Server, therefore this node can be omitted.

List Renderer (<listrenderer>)

Currently only the default list renderer is delivered with FSI Server, therefore

this node can be omitted.

OutputEffects (<outputeffects>)

This optional section provides the ability to define output effects that will be applied to every image accessed via this profile. Effects are defined by a list of <effects> tags containing effect commands using the same syntax as in image URLs. → See *chapter 7*.

OutputCompression (<outputcompression>)

Defines the output format and options for responses to image requests.

Compressor

Specifies the output file format, currently JPG, PNG and SWF are supported.

Quality

Defines the default jpeg compression quality. Please note that requests containing the quality parameter can override this setting.

4. Licensing

An unlicensed copy of FSI Server is fully functional but all images delivered will contain watermarks.

To purchase a licence please contact your local reseller or NeptuneLabs directly.

A licence is installed by copying it into a file called **licence.xml**, placing this into the → [CONFIG DIRECTORY]/`config` folder in your FSI Server installation directory and restarting your application server.

Depending on the licence obtained, limitations apply according to the domain name used for accessing the server or the number of images hosted on the server. A licence can also contain an expiration date. If any of these limitations are exceeded, all images delivered by the server will again contain the watermark. In this case contact your reseller or NeptuneLabs to upgrade or extend your licence.

Please note: The FSI Viewer and used Add-Ons or Plug-Ins must be licensed separately. Details on licensing FSI Viewer can be found in the FSI Viewer Manual.

5. Logging

FSI Server uses the Java log4J Logging framework. The configuration file for the used loggers is called **log4j.xml** and can be found in the → [CONFIG DIRECTORY]/`config` directory of the applications installation directory. The server uses three loggers that can be configured independently. The first logs the activity of the importer component, the second logs the server component and the third logs activity of the webinterface. By default all three loggers will write to rotating files in the application servers log directory. For more information on log4j and how to change the logging configuration please see the log4j website at → <http://logging.apache.org/log4j/>.

6. Templates

Responses to metadata requests for directories and images are rendered using templates. The response type and the content of the response can be modified by editing the templates or adding new templates. The templates are located in `WEB-INF/templates` in the FSI Server installation directory. A template must end with the suffix `.ftl` and is addressed by the **tpl** parameter in metadata requests (see part II, section 7.7). The templates are rendered using the freemarker template engine, documented at → <http://www.freemarker.org/docs/>.

Modifying the default templates is not recommended as the webinterface and FSI Viewer rely on correctly formatted meta data responses.

NeptuneLabs FSI Server

Part II • User Documentation

1. Getting Started

FSI Server includes a webinterface allowing easy access to most of the servers features. Due to the intuitive and technically advanced interface, browsing and managing images is simplified. The interface provides the full functionality needed to assist in publishing images, showcases and catalogs and also allows configuring the real-time image effects. To get started, simply open up the interface by pointing your browser to

→ <http://your.fsi-server.com/fsi>.

FSI Server is set up to provide real-time web access to images placed within a directory structure. Unlike any other imaging server, the server component merely requires read access to the directory. This guarantees the integrity of the original image files. This source directory is monitored and any new images are imported into FSI Servers internal storage retaining the directory structure. This internal storage consists of copies of the original files, optimized for fast access and real-time delivery to the web. This technique ensures high scalability and allows FSI Server to be used for high performance, heavy traffic websites.

Once configured the FSI Server does not require any further administration, the internal storage is set up automatically and will always be synchronized with the source directory. Modifications to the source directory via local file access, FTP Upload or the webinterface are immediately mapped to the internal storage.

2. Interface Overview

This section gives a brief overview of the web interface and its components. The interface is designed to provide a quick and intuitive access to the managed images and uses the modern AJAX technology. A list of compatible browsers can be found in → *part I - section 1*.

3. Logging In

When accessing the interface URL at → <http://your.fsi-server.com.com/fsi> a login screen and a language option will be displayed. Users need to authenticate themselves by using a combination of username and password. The default username and password combination is admin/admin, but it is strongly recommended to change the password prior to using FSI Server. For details on managing users please see → *Section 3 in part I - administrator documentation*.

The screenshot displays the NeptuneLabs - FSI Server interface. At the top left, there is a search bar with the text "Enter search query here" and a "Search" button. Below it is a "Folder Tree Menu" showing a hierarchical view of folders: "Admin Center", "Search Results", "Favorite Folders", "images", "3D product shots", "product shots", "catalogues", "sample catalogue", "sample 1", "sample 2", "sample 3", "sample 4", and "sample 5".

On the right side, there is a "View Bar" with a "Sort item by" dropdown menu set to "Name". Below it is a "Location Bar" with a search input field. At the bottom right, there is a "Thumbnail Canvas" displaying a grid of 15 image thumbnails, each labeled with a file name such as "page001.tif" through "page015.tif".

At the bottom of the interface, there is a "Tool Bar" with various icons and a "Log out" button. The "Log out" button features the FSI Server logo and a red power icon. The tool bar includes options like "Publish to Web", "FSI Viewer", "Common Tools", "No Navigator", "Minimal Interface", "No user interface", "FSI Viewer 360", "FSI Viewer 3D", "FSI Viewer 360 + 3D", "FSI Showcase", "HTML Image", "Item Properties", "Upload Images", "Download Image", "File and Folder Tools", and "Server Status".

At the bottom right corner, there is a "Status Bar" displaying the text: "Images: 134 - Last Changes: Tuesday, January 27, 2009 2:23:19 PM - Range: 1 - 15". Below the status bar, the copyright notice "© NeptuneLabs, 2009" is visible.

3.1 Search Bar

The "Search Bar" contains a textfield for entering search phrases and a submit button. A search phrase can be submitted by clicking on the submit button or simply by pressing enter. After submitting a search the results will be displayed in the thumbnail canvas and the search phrase is stored in the "Search Results" special folder in the folder tree. For details on search phrases and searching in general please see → *section 3.5*.

3.2 View Bar

The "View Bar" contains buttons to customize the appearance of the thumbnail canvas. The left button group switches between standard and page view mode whereas the right slider changes the thumbnail size. The dropdown selector in the center allows to change the sorting order of the thumbnails. If the Cooliris browser plug-in is installed, an additional button allows to activate a Cooliris wall, showing the current active folder.

3.3 Log Out

Clicking the log out button closes the current session and takes the user back to the login screen.

3.4 Folder Tree Menu

The control buttons for the folder tree are located in the "Folder Tree Menu". There are buttons for adding and removing favorites as well as refreshing the current view. Most commands are also available in the folder trees context menu.

3.5 Location Bar

Similar to a browsers location bar it shows the currently viewed position and allows entering a different location (folder) to view. Manually typing a new location is supported by an auto complete function. Additional buttons to go back and forward to the previous/next folder as well a button to jump to the parent folder make navigating easier.

3.6 Folder Tree

The "Folder Tree" shows the directory structure of the data stored on the imaging server. Branches can be expanded or collapsed by clicking on the + or – buttons in front of the folder name.

Selecting a folder will show the folders contents in the thumbnail canvas.

Two special folders contain shortcuts to favorites and the search history.

Right-clicking on an entry will display a context menu providing access to available commands for the selected folder.

3.7 Thumbnail Canvas

The "Thumbnail Canvas" shows the contents of the currently selected folder or the search results. Depending on the view mode directories and images or only images are displayed. Hovering above an item in the canvas will show a tool tip containing a brief summary of the item. Double clicking a folder item will change the current directory to that folder and double clicking an image item will open the image details popup described in →2.11. Items can be selected in order to perform operations with them. A single item is selected using a left click. Multiple items are selected by dragging open a box around them or by holding SHIFT and clicking another item. Items can be added to a selection by holding the CTRL key. As in the folder tree, right clicking will open a context menu containing the available operations.

3.8 Tool Bar

The "Tool Bar" mainly contains operations that can be performed with the selected items or in the current directory. Operations are enabled or disabled depending on the selection. The first tab contains the publishing options described in chapters →4 and →5. The second tab contains a button to access the image details popup for a single selected image. The next three tabs, "Upload", "Download" and "File and Folder Tools" provide access to the image managing options described in →chapter 3. The last tab shows the current server status for informational purposes.

3.9 Status Bar

The "Status Bar" displays additional information related to the curenly folder including the last modified date, the number of images, directories and the currently visible range of items. It also displays the currently logged in user in the bottom right corner.

3.10 Image Details Popup

The image details popup can be accessed by selecting the "Image Properties" entry from an image context menu, by clicking the "Show Details"-Button in the Tool Bar or by double clicking on an image in the Thumbnail Canvas.

The image details popup contains two tabs. The first tab shows the image in an FSI Viewer instance and the second tab provides access to all the meta data available for the selected image.



4. Managing Images

The FSI Server interface does not only assist in publishing data stored on the server, it is also an easy and comfortable way to manage images in terms of uploading, deleting, moving or renaming. An extensive search function also helps to organize large amounts of images.

4.1 Uploading

Images can be uploaded to the current directory by selecting one of the upload buttons from the Tool Bar. The first upload button enables a flash based upload function.

The system file chooser dialog used by this upload method, allows to select multiple files, which can be uploaded at the same time. Due to implementation issues in Flash Player the flash based upload does not work on browsers other than Internet Explorer if you are:

- a) connected to the internet via a proxy server
- b) accessing FSI Server interface via HTTPS

In these cases you should use the HTML/Javascript based upload instead which does not allow selecting multiple files to upload at a time though. Nevertheless you can add multiple images or ZIP archives to upload one after the other before uploading the selected files.

In addition to uploading single images, ZIP files can also be uploaded. These will then be unpacked to the upload directory and the images will be imported into the internal storage retaining the directory structure from the ZIP file.

4.2 Downloading

Downloading images is limited to either downloading the original image file or using the real-time image delivery. A download other than the original is therefore a standard single source imaging request directed at FSI Server. The output formats are limited to jpg or png and the maximum size for such a download is 3000 x 3000 px.

4.3 Deleting

Folders and images can be deleted by selecting the appropriate item in the thumbnail canvas and opening "File and Folder Tools" in the toolbar, then by clicking delete or by opening the context menu of the item and then clicking delete. Please note that deleting is irreversible and deleted items cannot be restored.

4.4 Moving and Renaming

Folders and images can be renamed by selecting the item in the thumbnail canvas and choosing "Rename" from the toolbar or the context menu.

Moving objects from one folder to another is accomplished by selecting the item, choosing "Cut", changing to the new folder and selecting "Paste". Please note that moving or renaming a folder containing lots of files or subfolders, requires a restructure of the internal storage and therefore might take a while.

4.5 Searching

FSI Servers search provides an easy way to find specific images, especially with a large amount of images on the server. The search function not only allows a full text search in the image meta data, but also allows image specific search such as looking for images with a certain width or height.

4.5.1 Full Text keyword search

By simply typing a keyword into the search field a full text search across the complete image meta data is performed. This includes all textual fields in the file specific meta data (the filename) as well as the IPTC and EXIF meta data of all currently imported images.

The full text search supports two wildcards: the question mark (?) stands for one single arbitrary character and the asterisk (*) stands for any number or arbitrary characters. Search terms may not begin with wildcards. The search can be limited to a single textual field by preceding the keyword with the fieldname followed by a colon, for example 'iptc.caption:holiday'

will show all images having the word 'holiday' in the IPTC Caption field.
→ *Appendix D.1* contains the full list of supported fields.

4.5.2 Comparative search

In addition to the full text search a comparative search is supported for a set of numerical meta data fields and fields containing dates. The comparative search always begins with a field name (see table below for a full list of supported field names). The value can be prepended with a smaller than (<) or larger than (>) prefix. If the prefix is missing, an exact match is required.

This for example allows searching for images with a minimum height (file.height:>3000) or images older than a specific date (iptc.date_created:<1.1.2009). → *Appendix D.2* contains the full list of supported fields.

Note on date searches: the given date can be provided in any standard format, independent of any country and/or regional settings. The date value may not include any spaces between the figures, as these would be treated as token separators.

4.5.3 Search modifiers

All entered keywords or phrases are optional by default. This means if two keywords are provided the result set will contain images having at least one the keywords in their metadata.

To perform searches with mandatory keywords precede the phrase with a plus (+). To exclude images from the results which match a certain condition precede the phrase with a minus (-).

5. Publishing Single Images

The publishing dialogs of FSI Servers interface provide assistance in generating image URLs or HTML code snippets, which can then be integrated into web pages.

A single image can be presented as a fixed size image in the end users browser or as an interactive zoomable image using FSI Viewer. The publishing dialogs can be accessed by selecting a single image in the thumbnail canvas and choosing a "Publish as HTML image" or "Publish as FSI Viewer" option from the toolbar.

5.1 HTML Image

The "Publish as HTML Image" helps assembling a URL, which can be used directly in an HTML img tag. The publishing dialog consists of four tabs allowing the selection of an image range, setting the resulting size and output effects and showing a preview of the resulting image.

Image URLs can also be assembled manually. → *Appendix B* contains a complete server URL command reference.

Publish Tab

The "Publish Tab" allows setting the resulting image size, the image format and in case of jpeg, the quality. It also contains a text field showing the HTML img tag, which can be integrated into webpage.

Effect Tab

The "Effect Tab" allows adding real-time image effects to the resulting image. Effects can be added by entering the effect parameters and then adding them in the effect list. Once in the effect list, effects can be dragged to change their position in the list, modifying the order they are applied.

Crop Tab

The "Crop Tab" consists of a preview image with the possibility of selecting the image range. If a range is selected, only the selection will be published.

Preview Tab

The "Preview Tab" shows a preview of the published image, with all effects and parameters applied. The preview size is limited to 700 px width.

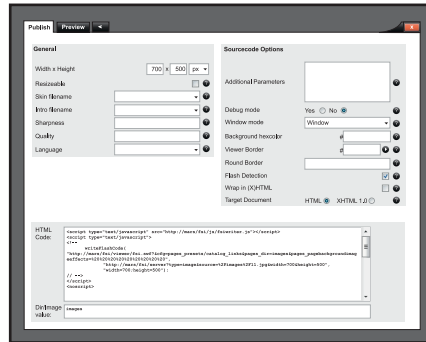
5.2 FSI Viewer

The "Publish as FSI Viewer" section allows choosing between four commonly used presets. Choosing a preset will open the FSI Viewer publishing dialog that provides access to all FSI Viewer parameters and plug-ins.

The publishing dialog contains five tabs grouping the FSI Viewer publishing options and parameters. By default only the two most commonly used tabs are directly accessible. The remaining tabs can be made visible using the expand tabs (>) button in the tab selection area.

Publish Tab

The "Publish Tab" provides access to the basic FSI Viewer and HTML source code options like Viewer size and language. It also contains a text field containing the HTML code snip assembled from all the settings made in the various tabs.



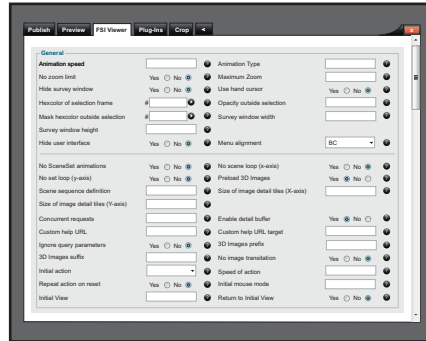
Preview Tab

The "Preview Tab" shows an FSI Viewer instance using the assembled code snip from the Publish Tab. The FSI Viewer looks exactly as if it were integrated into a web page, which is limited to 700 px width. An "Open in Popup" button allows previewing the FSI Viewer in a separate window where the size limitation no longer applies.



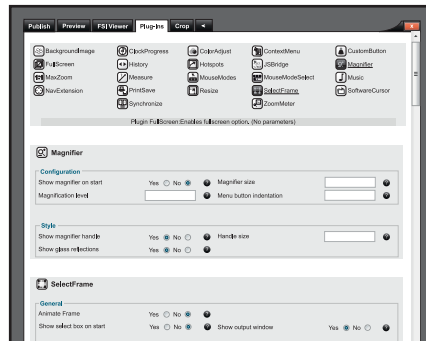
FSI Viewer Tab

The "FSI Viewer Tab" provides access to the main viewer configuration parameters. The parameters mainly affect the behavior and appearance of the Viewer. Like in all the other tabs in the publishing section a short description is available for every parameter by clicking on the question button next to the parameter input field. For a more detailed help on FSI Viewer configuration parameters please see the FSI Viewer manual.



Plug-Ins Tab

The "Plug-Ins Tab" not only allows to activate and deactivate the various FSI Viewer plug-ins, it also shows all configuration parameters of the activated plug-ins. Clearly arranged for quick access, the tab is split up in boxes. The first box contains a list of available plug-ins. Activating a plug-in by clicking on the plug-in icon will show a further box containing this plug-in parameters. For a detailed description of all available FSI Viewer plug-ins see the FSI Viewer manual.



Crop Tab

As in the "Publish as HTML Image" section, the "Crop Tab" allows selecting an image range which will then be published in the FSI Viewer

6. Publishing Image Collections

Image collections can be published by placing the desired images into a folder and publishing the folder. Collections are used for 3D object presentation, FSI Pages and FSI Showcase. A folder can be published by navigating into the folder and selecting an FSI Viewer 360, Pages or Showcase preset from the Tool Bar.

6.1 FSI Viewer 360

Publishing a collection as FSI Viewer 360 will create an FSI Viewer instance, displaying a rotatable view of an object composed by all images in the collection. The publishing dialog is identical to publishing a single image. Please refer to section 4.2. for details.

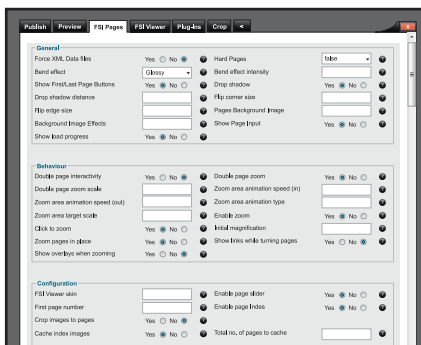
6.2 FSI Pages and FSI Pages mobile

Choosing an FSI Pages preset for a collection will activate the FSI Pages add-on. The collection will be presented as a book or catalog. In the FSI Pages publishing dialog, the "Plug-Ins Tab" is extended by seven additional plug-ins and a new tab called "FSI Pages" is available after clicking on the expand tabs button.

FSI Pages Tab

The "Pages Tab" provides quick access to all FSI Pages specific parameters. For details on the various configuration options please see the FSI Pages section in the FSI Viewer manual.

The FSI Pages publishing interface is also used to publish image collections with FSI Pages mobile. The publish tab contains code snips for both FSI Pages and FSI Pages mobile.

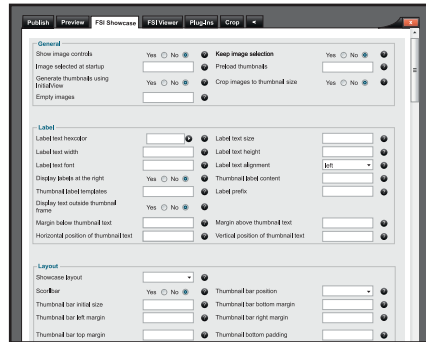


6.3 FSI Showcase

Choosing an FSI Showcase preset when publishing a collection will activate the FSI Showcase add-on. A showcase allows to present multiple images in a single FSI Viewer instance as a kind of image gallery. The FSI Showcase publishing dialog adds a new tab to the common FSI Viewer publishing dialog.

FSI Showcase Tab

The "FSI Showcase Tab" provides quick access to all FSI Showcase specific parameters. For a detailed description of the configuration options please see the FSI Showcase section in the FSI Viewer manual.



7. Publishing Stereoscopic images

FSI Server supports real-time generation of stereoscopic 3D images, since the latest version 2. The only things required are two source images of the same object, taken from slightly different angles. The two source images can be published as one plain stereoscopic image or as an FSI Viewer instance, allowing the user to zoom into the image. Two sets of source images can be published as an FSI Viewer 360° spin instance, if they are placed into separate directories.

7.1 Publishing a single stereoscopic image

Selecting two images with the same dimensions in the thumbnail canvas enables the "Publish as 3D HTML image" option in the toolbar. Selecting this option brings up a publishing dialog similar to that used for publishing a standard HTML image (see 4.1). The dialog has three additional parameter fields in the publish tab, allowing to select the type of 3D glasses by specifying the lens colors for the left and right eye as well as choosing a desaturation effect to increase the 3D impression.

7.2 Publishing a stereoscopic image in FSI Viewer

Selecting two images with the same dimensions in the thumbnail canvas also enables the "Publish as FSI Viewer 3D" option in the toolbar. Here too, the resulting publishing dialog will be identical to the standard "Publish as FSI Viewer" dialog with the three additional fields described in the chapter above.

7.3 Publishing stereoscopic images as FSI Viewer 360

Selecting two directories containing the same number of images in the thumbnail canvas will enable the "Publish as FSI Viewer 360 + 3D" option in the toolbar. This option opens the FSI Viewer 360 publishing dialog with the additional parameter fields as described above.

8. Server Addressing

The FSI Server is addressed using standard HTTP and HTTP GET query parameter to specify the request.

The main server URL will always be in the form `http://your.fsi-server.com/fsi/server?` followed by the desired parameter. If the application is deployed under a name different to "fsi" the above URL needs to be adapted accordingly. The following sections describe the available parameters in detail and give examples on their usage. Please note that not all parameter combinations are possible. In addition to the following descriptions. → *Appendix B* contains a server addressing quick reference. The parameters described in this chapter apply to the default renderers. Other renderers might not support all the parameters listed here or rely on additional parameters. More on renderers can be found in the OpenAPI manual.

8.1 Source

The source parameter describes the accessed resource in form of a path. This can be either an image or a directory. The source parameter is mandatory.

8.2 Type

The mandatory type parameter defines the type of response expected. Available values are *image*, *info* and *list*. Image and info can be used if the source parameter defines an image file. Depending on the profile used (see → 6.9) an image request will return a jpg, png or swf. Info and list requests provide access to image and directory metadata. The response type of these two depends on the used template (see → 6.7).

8.3 Width and Height

At least one of the parameters width or height must be provided for image requests. The value is interpreted as pixels and the result will be an image scaled to fit into the requested dimension. Depending on the profile used the aspect ratio is preserved, so if width and height are specified the resulting image may be smaller than the requested size.

8.4 Rect

The optional rect parameter can be used in image requests. It allows cropping images prior to scaling. The value is expected to consist of four comma-separated float values. The first two values define the top-left corner and the other two values define the bottom-right corner of the crop area. All values are expected to be normalized, this means the valid values range from 0 to 1 where 0 is left or top and 1 is right or bottom.

8.5 Left, Top, Right, Bottom

The parameters left, top, right, bottom are a more human readable version of the rect parameter described above. The possible values are identical (0 to 1).

8.6 Quality

Used in image requests that return JPEG images this optional parameter defines the JPEG compression quality. The default value depends on the profile settings.

8.7 Tpl

The tpl parameter defines the template used to render info and list responses. For more on templates see → *section 7 in part I*.

8.8 Effects

The effects parameter allows applying real-time image effects to an image. The effects parameter value is a comma-separated list containing the effect name followed by brackets containing effect-specific parameters. The effects will be applied in the order they are listed. For a detailed description of the available effects and the effects-specific parameters please see → *section 8*.

8.9 Profile

Used in all request types this optional parameter defines the server profiles used to build the response. If omitted the default profile will be used. The default profile is defined by the server settings. For more information on profiles see → *section 3 in part I*.

9. Real-time Image Effects

The FSI Server provides various effects, which are available in real-time and can be applied to the images before delivery.

This section describes and visualizes the included effects using an example image. The left image corresponds to the original and the right image shows the applied effect(s).

Blur	
Radius	(Float) $\geq 0.0 \leq 100.0$
Example	blur(50)
Version	1.0

Applies a Gaussian blur effect to the image and therefore reduces image noise and detail levels.



Sharpen	
Amount	(Float) $\geq 0.0 \leq 100.0$
Example	sharpen(50)
Version	1.0

Sharpens the image using a Gaussian unsharpen algorithm. Especially details and contours will be highlighted.



Glow	
Amount	(Float) $\geq 0.0 \leq 100.0$
Example	glow(25)
Version	1.0

Adds a glow effect to the image by enlarging and summarizing all bright areas, so you have the impression the image shines.



Pixelize	
Size	(Int) ≥ 0
Example	pixelize(30)
Version	1.0

Creates a pixel effect by reducing the image details. The pixel colors adapt to the most common used color around each pixel, resulting in a tile effect.



Flip	
Value	(Enum) Horizontal, Vertical
Example	flip(vertical)
Version	1.0

Flips the image horizontally or vertically.



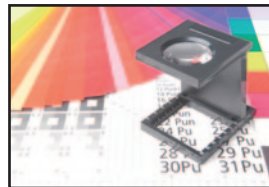
Distort	
Value	4 normalized xy coordinate pairs describing the new corner positions.
Example	distort(0.1,0.1,0.9,0.2,1,1,0,1)
Version	1.0

Performs a perspective distortion by moving the image corners to the new specified coordinates. The first two values are the upper left corner, based on a value of 0-1 on the X-and Y-axis. The following corners are to be entered clockwise.



Brightness	
Value	(Int) $\geq -100 \leq 100$
Example	brightness(20)
Version	1.0

Increases or reduces the images brightness.



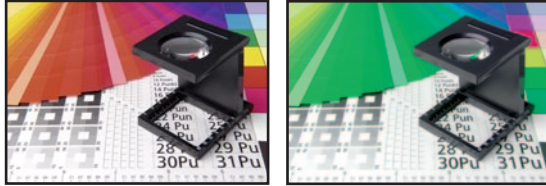
Contrast	
Value	(Int) $\geq -100 \leq 100$
Example	contrast(20)
Version	1.0

Increases or reduces the image contrast.



ChannelExchange	
Mode	(Enum) RGB, RBG, BGR, BRG, GRB, GBR
Example	channelexchange(grb)
Version	1.0

Swaps color channels.



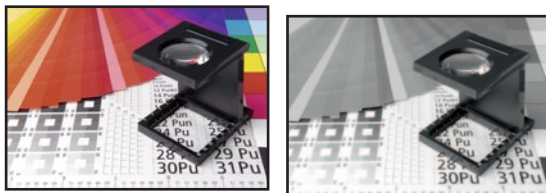
Colorize	
Hue	(Float) $\geq 0.0 \leq 360.0$
Saturation	(Float) $\geq 0.0 \leq 100.0$
Lightness	(Float) $\geq 0.0 \leq 100.0$
Example	colorize(170,80,10)
Version	1.0

Applies a colorize effect to the image. This makes it possible to give a general color impression to the image.



Desaturate	
Mode	(Enum) Average, Lightness, Luminosity, Saturation
Example	desaturate(lightness)
Version	1.0

Desaturates the specified image property.



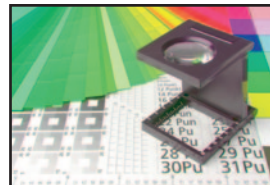
Gamma	
Value	(Float) $\geq 0.1 \leq 9.99$
Example	gamma(6.0)
Version	1.0

Reduces or increases the image gamma.



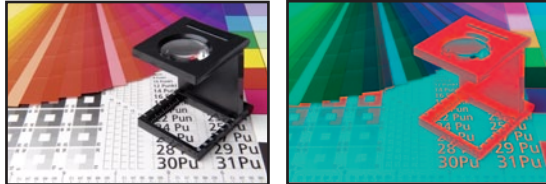
HSL	
Hue	(Float) $\geq -180 \leq 180$
Saturation	(Float) $\geq -100 \leq 100$
Lightness	(Float) $\geq -100 \leq 100$
PrimaryColor	(Enum) All, Red, Yellow, Green, Cyan, Blue, Magenta
Example	hsl(90,80,30,All)
Version	1.0

Allows increase or decrease of the hue, saturation and lightness of a specified image channel.



Invert	
Mode	(Enum) Brightness, All, Red, Green, Blue, Alpha
Example	invert(red)
Version	1.0

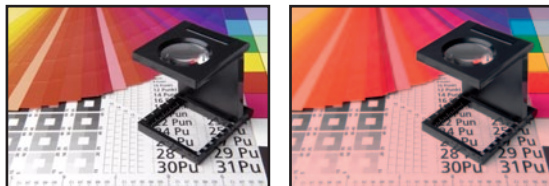
Inverts one channel, all channels or the brightness of the image.



ColorOverlay	
Blendmode	Enum Blendmodes (Normal, Dissolve, Exclusion, Difference, Hardmix, PinLight, LinearLight, VividLight, HardLight, SoftLight, Overlay, LighterColor, LinearDodgeAdd, ColorDodge, Screen, Lighten, DarkenColor, LinearBurn, ColorBurn, Multiply, Darken, Divide, GrainExtract, GrainMerge, Subtract, LuminosityHSL, ColorHSL, SaturationHSL, HueHSL, LuminosityHSB, ColorHSB, SaturationHSB, HueHSB.)
RGB	(RGB INT) $\geq 0x000000 \leq 0xffffffff$
Opacity	(float) $\geq 0.0 \leq 100.0$
Example	coloroverlay(VividLight,0x300000,20)
Version	1.0

Applies a color overlay to the image using the specified color and blend mode.

By using one or perhaps several different effects, many changes in coloration can be achieved.



Posterize	
Level	(Int) $\geq 0 \leq 255$
Example	posterize(11)
Version	1.0

Applies a posterize effect, reducing the image to a limited number of color levels. Posterize is a separation of the tonal values with a simultaneous tonal value reduction.



Threshold	
LevelLow	(Int) $\geq 0 \leq 255$
LevelHigh	(Int) $\geq 0 \leq 255$
Color1	(RGB INT) $\geq 0x00000000 \leq 0xffffffff$
Color2	(RGB INT) $\geq 0x00000000 \leq 0xffffffff$
Example	threshold(10,200,0x00471698,0x00fadb07)
Version	1.0

Reduces the color range of the images by applying a threshold filter.



Transparency	
Value	(Int) >0 <= 100
Example	transparency(30)
Version	1.0

Adds transparency to the image. If the image contains an alpha channel it will be strengthened. If there is no alpha channel in the image, then one will be created.



Matte	
Value	(Hex RGB)
Example	matte(2cb4ac)
Version	1.0

Removes transparency and renders all translucent pixel against the matte color.



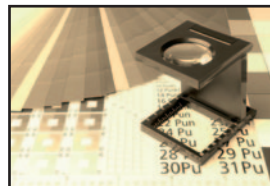
Noise	
Amount	(Int) >0 <= 100
Strength	(Int) >0 <= 100
Monochrome	(Boolean)
Example	noise(60,80,true)
Version	1.0

Adds noise to the image.



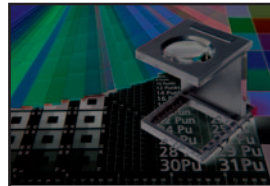
Sepia	
Example	sepia()
Version	1.0

Makes the image look like a sepia toned photo.



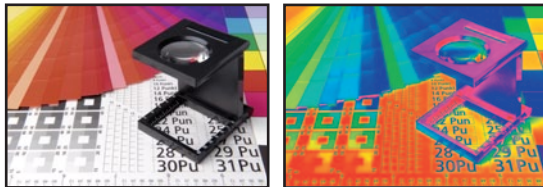
Solarize	
Example	solarize()
Version	1.0

Applies a solarize effect that negates all pixels above a certain value. Solarization allows the alienation of images.



GradientMap	
Mode	(Enum) Average, Lightness, Luminosity
Value	(HexColor) ARGB, [ARGB], ...
Example	gradientmap (luminosity,ff0000,ff00ff,0000ff,00ffff,00ff00,ffffff00,ffff00,ff0000)
Version	1.0

Maps a defined gradient over a grayed image.



Appendix

Image Formats

Supported Image Formats are:
PNG, TIFF*, JPEG, BMP, GIF, FPX

* special features like layer effects are currently not supported

A Supported Input/Output Formats

1. Input Formats

Format	Description
PNG	Portable Network Graphics files are fully supported, including alpha channels. The maximum image size depends on the amount of memory available.
JPEG	Joint Photographic Experts Group files are fully supported. The maximum image size depends on the amount of memory available.
TIFF	Tagged Image File Format files are fully supported, including pyramid tiffs. The maximum image size is limited to 2GB assuming the necessary hard disk capacity is provided.
BMP	Bitmap file format is fully supported. The maximum image size depends on the amount of memory available.
FPX	FlashPix file format is fully supported. The maximum image size depends on the amount of memory available.
GIF	Graphics Interchange Format is fully supported. The maximum image size depends on the amount of memory available.

2. Output Formats

Format	Description
JPEG	The default output format providing the additional quality parameter to specify image compression quality
PNG	The png output supports alpha channels by default.
SWF	A special format wrapping image data into an swf for image presentation within flash movies.

B Server Addressing Quick Reference

1. Image Requests

?type=image&source=path/to/image

Parameter	Description	Mandatory	Default
Width	Width/height of the resulting image	yes (one)	server defaults
Height	Width/height of the resulting image	yes (one)	server defaults
Rect	Crop to a given range	no	0,0,1,1
Left	Crop to a given range	no	left=0
Right	Crop to a given range	no	right=1
Top	Crop to a given range	no	top=0
Bottom	Crop to a given range	no	bottom=1
Quality	Jpeg compression quality	no	depends on profile
Effects	Real-time image effects	no	none
Profile	Used profile	no	depends on settings

2. Info Requests

?type=info&source=path/to/image

Parameter	Description	Mandatory	Default
Tpl	Defines the template used to render the response	no	depends on profile
Profile	Used profile	no	depends on settings

3. List Requests

?type=list&source=path/to/directory

Parameter	Description	Mandatory	Default
Tpl	Defines the template used to render the response	no	depends on profile
Limit	Limits the amount of entries in the result set	no	all results
Profile	Used profile	no	depends on settings

4. Search Requests

?type=search

Parameter	Description	Mandatory	Default
source	Path in which to search	yes when not authenticated	empty
query	Search phrase	yes	–

?type=simplesearch

Parameter	Description	Mandatory	Default
source	Path in which to search	no	empty
context	Comma-separated list of fields to search (see Appendix D for field names)	yes	–
query	Search phrase	yes	–

C Tomcat and JVM Settings

1. Java Virtual Machine Settings

The optimal settings for the Java virtual machine depend on the hardware running FSI Server and any other application sharing the hardware resources. Generally speaking, increasing the memory available in the virtual machine will increase FSI Servers performance as larger images can be converted in memory and more objects can be cached.

The recommended setting for Oracle Java SE is:

```
-XX:+UseConcMarkSweepGC  
-XX:+UseParNewGC  
-XX:+UseBiasedLocking  
-XX:+CMSClassUnloadingEnabled  
-XX:+CMSParallelRemarkEnabled  
-Xms RAM Size / 2 - 500 MB  
-Xmx RAM Size / 2 - 500 MB
```

2. AJP & JK

If FSI Server is set up in an application server, reachable via AJP only, please ensure the Proxy Pass Reverse Settings are included in the AJP configuration, as the webinterface will not be usable with incorrect or missing AJP settings.

```
<IfModule mod_proxy_ajp.c>  
  <Location /server>  
    ProxyPass ajp://localhost:8009/fsi  
    ProxyPassReverse ajp://localhost:8009/fsi  
    ProxyPassReverseCookiePath /fsi /server  
  </Location>  
</IfModule>
```


D Search field reference

1. Textual search field

Fieldname	Description
file.name	the image filenames
iptc.caption	The contents of the IPTC 'Caption' field
iptc.writer	The contents of the IPTC Writer/Editor field
iptc.headline	The contents of the IPTC Headline field
iptc.special_instructions	The contents of the IPTC Special Instructions field
iptc.by_line	The contents of the IPTC By-line field
iptc.by_line_title	The contents of the IPTC By-line Title field
iptc.credit	The contents of the IPTC Credit field
iptc.source	The contents of the IPTC Source field
iptc.object_name	The contents of the IPTC Object Name field
iptc.city	The contents of the IPTC City field
iptc.province_or_state	The contents of the IPTC Province/State field
iptc.country_or_primary_location	The contents of the IPTC Country/Primary Location field
iptc.original_transmission_reference	The contents of the IPTC Original Transmission Reference field
iptc.category	The contents of the IPTC Category field
iptc.supplemental_categories	The contents of the IPTC Supplemental Categories field
iptc.keywords	The contents of the IPTC Keywords field
iptc.copyright_notice	The contents of the IPTC Copyright Notice field
iptc.originating_program	The contents of the IPTC Originating Program field
iptc.fsi_search_data	The contents of the IPTC FSI Search Data field
exif.exposuretime	The contents of the EXIF Exposure Time field
exif.fnumber	The contents of the EXIF Fnumber field
exif.exposureprogram	The contents of the EXIF Exposure Programm field
exif.isospeedratings	The contents of the EXIF ISO Speed Ratings field
exif.subjectdistance	The contents of the EXIF Subject Distance field

2. Comparative search fields

Fieldname	Description	Type
file.size	The file size in bytes	Number
file.width	The image width in pixels	Number
file.height	The image height in pixels	Number
file.lastmodified	The last modified date of the original image file	Date
iptc.release_date	The contents of the IPTC 'Release Date' field	Date
iptc.date_created	The contents of the IPTC 'Date Created' field	Date

E Updating from FSI Server 1 to 2

FSI Server introduces a couple of new features that require updating the configuration files. Most configuration options have remained the same and nearly all new parameters have usable default values, so the necessary changes have been reduced to a minimum. If you want to reuse your old configuration files, then make sure to perform the changes described in this chapter in order to achieve the same behavior as expected.

1. Update using the Windows installer version

The Windows installer will automatically detect an old version of FSI Server and will update the configuration as needed. A manual adaptation of the to the new parameters should not be necessary.

2. Updating the platform independent version

The best way update the platform independent version of FSI Server is to make a copy of the configuration files, undeploy the old FSI Server application, deploy the new version and copy the configuration files into the new installation, applying the changes described below. If the you uploaded custom images into the samples directory, please be sure to make a copy of WEB-INF/samples in your old installation. The same applies to the storage directory: If your storage is located in the WEB-INF directory, then you need to copy of move it out prior to undeploying the old FSI Server version.

3. User configuration

FSI Server 2 introduces advanced user and permission configuration options, to allow ASP services. Unlicensed FSI Server installations behave like ASP Editions, to allow full testing. Customers with valid licences for Non-ASP Editions (Basic, Standard or Enterprise) can use the configuration files from FSI Server 1 without applying any changes. Customers with ASP licences need to modify their users.xml and give their current accounts administrator permissions if they want the same behavior as in version 1. In order to do this, the attribute accesslevel with a value of "admin" needs to be added to each user entry. The new entries then look like the following example:

```
<user name="admin" hash="8c6976e58...f6f2128736918" accesslevel="admin" />
```

4. Import Scanner

The scanning process used by the importer now has more configuration options, so please be sure to add the following section into the importer.xml file:

```
<scanner enabled="true">
    <sorting>true</sorting>
</scanner>
```

5. Search database configuration

The search database has been modularized to prepare future plug-ins for other databases. Please remove all tags beginning with search_ from the **settings.xml** and the **importer.xml** file and add the following new section after the application section in both files:

```
<search enabled="true">
    <type>h2</type>
    <database>search</database>
    <username>sa</username>
    <password>sa</password>
</search>
```

All other configuration files can be reused without any alterations. Default values will be applied where necessary and will enforce the same behavior as in version 1. ASP customers should be sure to have read section 3.2 (Administrator Documentation), which describes the user and permission management before trying to configure any user accounts.

Index

Application	18	Sepia	51
Imagethreads	19	Sharpen	43
Importstrategy.....	19	Solarize	51
Maxcpu	19	Threshold	49
Storage	18	Transparency	50
Tempdirectory.....	20	Effect Tab.....	35
Bottom.....	41	Examples.....	13
Configuration.....	11	Folder Tree	31
Configuring the importer compo- nent.....	11	Folder Tree Menu	30
Configuring the real-time image server component	14	FSI Pages	38
Crop Tab	35, 37	FSI Pages Tab.....	38
Deleting	33	FSI Showcase	39
Downloading	33	FSI Showcase Tab.....	39
Effects.....	41	FSI Viewer.....	35
Blur.....	43	FSI Viewer 3D.....	38
Brightness	45	FSI Viewer Tab.....	37
ChannelExchange.....	46	Groups	13, 23
Colorize.....	46	height	41
ColorOverlay	48	HTML Image.....	35
Contrast	45	Image Details Popup	32
Desaturate	46	Importer component setting ..	18
Distort.....	45	Importer profiles.....	22
Flip	44	Import Profiles.....	12, 13
Gamma	47	Info Renderer	25
Glow	44	Installation	9
GradientMap.....	52	Installation on Microsoft Win- dows.....	9
HSL.....	47	Interface Overview.....	28
Invert.....	48	Left.....	41
Matte	50	Licencing.....	27
Noise	51	List Renderer	25
Pixelize.....	44	Location Bar	30
Posterize	49	Location of the configuration files 18	

Logging	27	Searching	33
Logging In.....	28	Settings	
Log out	30	Colormanagementsystem	21
Managing Images	32	Formats.....	21
Moving and Renaming	33	Protected_dirs	21
Outputcompression	26	Protected_files	21
Outputeffects.....	26	Source	40
Parallelread.....	19	Standalone	20
Parallelreadwrite	19	Status Bar	31
Parallelwrite.....	19	System Requirements.....	9
Permission Sets.....	12, 23	Templates.....	27
Ping service	22	Thumbnail Canvas.....	31
Platform independant installation		Tool Bar	31
10		top.....	41
Plug-ins Tab.....	37	Tpl.....	41
Preconsiderations.....	11	Type.....	40
Prefetch	19	Uploading.....	32
Prefetchmaxsize.....	19	User Management.....	18
Preview Tab	35, 36	Users	24
Profile	42	Users, Groups.....	12
Profiles.....	25	View Bar	30
Conversion	22	Width.....	41
Enabled.....	22		
Origin.....	22		
Outputcompression	26		
Renderer	25		
Publishing Image Collections..	38		
Publishing Single Images	34		
Publish Tab.....	35, 36		
Quality	41		
Realtime image server compo-			
nent settings	24		
Rect.....	41		
Right.....	41		
Search Bar.....	30		

FSI Server

NeptuneLabs GmbH

Lagesche Str. 32
32657 Lemgo
Germany

Fon: +49 5261-28732-0
Fax: +49 5261-28732-29
eMail: info@neptunelabs.com
http: www.neptunelabs.com
www.fsi-server.com

No part of this manual, including the software described in it, may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form or by any means, except documentation kept by the purchaser for backup purposes, without the express written permission of NeptuneLabs.

Specifications and information contained in this manual are furnished for informational use only and are subject to change at any time without notice, and should not be construed as a commitment by NeptuneLabs. NeptuneLabs assumes no responsibility or liability for any errors or inaccuracies in this manual, including the software described in it.

Copyright 2009-2011 NeptuneLabs GmbH, Germany. All rights reserved.

www.fsi-viewer.com

Neptune
L a b s

Internet Application Development

NeptuneLabs GmbH

Lagesche Str. 32

32657 Lemgo

Germany

fon: + 49 5261 - 28732- 0

fax: + 49 5261 - 28732- 29

info@neptunelabs.com

www.neptunelabs.com