

FSI Cache

User Manual



Version 4.1

FSI Cache 4.1

NeptuneLabs GmbH
Lagesche Str. 32
D-32657 Lemgo
Germany

© 2009-2015 NeptuneLabs. All rights reserved.

Last updated: July 2011

FSI Cache Version:	4.1
Manual Revision:	004

All brands and product names are trademarks or registered trademarks of the respective producers. FSI Viewer, FSI Server and NeptuneLabs are registered trademarks of NeptuneLabs GmbH, Germany.

Table of Contents

1 Introduction	5
1.1 Preface	5
1.2 System Requirements	5
1.3 Compatibility	6
2 Installation	7
2.1 Windows Installer	7
2.2 Platform independent installer	7
3 Essentials	9
3.1 How it works	9
3.2 Profiles	9
3.3 Addressing	9
3.4 Keep the cache clean	10
4 Configuration	11
4.1 Configuration Parameter Quick Reference	11
4.2 Profile Settings	22
4.3 Basic Configuration	27
4.4 Remote Sync Service	27
4.5 Cache Control Service	28
4.6 DoS Protection (Enterprise Edition only)	29
4.7 Fallback Imaging Server	29
4.8 Balancing Imaging Servers (Enterprise Edition only)	30
4.9 Integrating FSI Cache into your IT environment	30
5 Configuration using Administrator	33
5.1 Connecting to your FSI Cache from FSI Administrator	33
5.2 Administrating FSI Cache	33
6 Advanced Setups	37
6.1 Balanced FSI Cache Servers	37
6.2 Cascaded FSI Cache instances	37

6.3 Example Setups	37
7 Using FSI Cache	41
7.1 Addressing	41
7.2 Image Ranges	41
7.3 Further server commands	42
7.4 Path encryption	42
8 Known Issues	45
8.1 Limit available disk space	45
8.2 Imaging server licenses and templates	45
8.3 eRez headers	45
Index	47

1 Introduction

1.1 Preface

FSI Cache is a web application designed to run in a Java servlet container. Its primary aim is to serve as a cache for one or more imaging servers. Imaging servers are optimized for processing image data and therefore generating content. They have hardware and system requirements quite different to those of a server meant for delivering content. In high performance environments with heavy traffic load on the servers, content delivery consumes immense system resources reducing image processing performance. In scenarios like this a cache is the best way to handle increasing traffic and optimize website performance.

1.2 System Requirements

The hardware requirements depend on the amount of images being hosted and the amount of hits and image requests the server has to handle. The minimum useful requirements are:

- **2 GHz Pentium 4 or better**
- **1 GB free memory (2 GB recommended)**
- **Operating System:**
 - A Sun Java 5 enabled operating system, such as:
 - Linux (Intel x86, Intel x86 64 Bit, PowerPC and others)
 - BSD
 - Microsoft Windows®, 2003, 2008, Vista or 7 (32 or 64 Bit)
 - Sun Solaris 10 (SPARC or Intel x86)
 - Mac OS X or Mac OS X Server
- **Web Application Server:**

A Java Servlet 2.3 compliant server. FSI Cache has been developed on and tested with Apache Tomcat > 5.5.27
- **Storage Requirements:**

The amount of disk space needed depends on the amount and the size of images you are going to cache. The FSI Cache itself will use about 40MB of hard disk space.

1.3 Compatibility

FSI Cache is primarily designed to serve as a cache for imaging servers. Currently supported imaging servers include FSI Server 1 & 2 by NeptuneLabs as well as the discontinued eRez Imaging Server 3-5 by YaWah.

2 Installation

As noted in the previous section FSI Cache runs as a WebApplication in a servlet container.

2.1 Windows Installer

The Windows installer package for FSI Server 2 includes a bundled FSI Cache as well as a bundled FSI Administrator. Please follow the on-screen instructions to perform the installation procedure. For further details please also see the FSI Server 2 manual.

2.2 Platform independent installer

To install FSI Cache on your running application server or servlet container simply deploy the delivered WebArchive (war-file). How to deploy an application depends on your server, in most cases it will be enough to drop the war-file into the servers webapps directory. Please consult your application servers manual for details.

If you have deployed FSI Cache successfully it might be necessary to tune your servlet containers settings in order for FSI Cache to run.

- **Allow write access**
FSI Cache needs to write the cached data to the disk. Therefore write access to the web applications directory is needed. If you are using Apache Tomcat you can allow write access by disabling the secure mode (TOMCAT_SECURITY=no)
- **Allow Encoded Slashes in URLs**
If you are using Apache Tomcat, please add "-Dorg.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODE" to the CATALINA_OPTS
- **Maximum Amount of memory**
It is recommendable to increase the at maximum amount of memory your application server can access. Especially when you are running more than one WebApplication or making use of the advanced features of FSI Cache. You can increase the amount of accessible memory by setting the Java VM heap size using the -Xmx parameter. The best value for this setting depends on the expected amount of traffic, your server hardware and the other applications running on your server.

3 Essentials

This chapter describes how FSI Cache works and explain all necessary terms to configure and operate FSI Cache successfully.

3.1 How it works



FSI Cache is meant to respond to all requests that would be directed at the imaging server if you weren't using a cache. Every request received by the cache is analyzed and FSI Cache will try to respond to the request by using cached data stored on the local disk. If necessary the underlying imaging server will be consulted to answer the request. FSI Cache has built in mechanisms to evaluate if the cached data is still valid or if images on the managing server have been deleted or modified. In the optimal setup, your imaging server must no longer be addressed directly as all image requests will be answered by FSI Cache.

3.2 Profiles

Apart from the general settings FSI Cache is configured using so called profiles. Every profile consists of a name and a set of configuration parameters that define how requests addressed to this profile are handled. The profile configuration includes settings like the imaging server(s) to be used and the maximum amount of files allowed in the cache.

3.3 Addressing

FSI Cache can be addressed in different ways. The Pure Edition expects request URLs to be formatted using simple path elements separated by slashes. A so formatted URL looks similar to the following:

```
http://your.fsi-server.com/fsicache/profile/path/to/image/width_height
```

In this example "your.fsi-server.com" stands for the hostname or IP-address of your server, "fsicache" is the name under which you deployed the application, "profile" denotes the name of the addressed profile (see previous section), "path/to/image" is the path to the image on the imaging server (including filename)

and "width" and "height" are numerical values for the size of the image requested. The Enterprise Edition of FSI Cache additionally supports requests using a query style format. The example request from above would then look as follows:

```
http://your.fsi-server.com/fsicache/profile?src=path/to/  
image&width=width&height=height
```

For more advanced issues on request formats and optional addressing parameters please refer to section 7.1

3.4 Keep the cache clean

One of the problems with caches is that the cached content may be out of date because the content on the originating server was deleted or modified. FSI Cache has several methods to verify the validity of the cached content and remove old unused data from the cache. Various options allow adapting the caching behavior of FSI Cache to your needs.

4 Configuration

Once installed you need to configure FSI Cache by editing the appropriate configuration parameters. The configuration can be performed by either editing the configuration files directly or by connection an FSI Administrator to your FSI Cache instance and using its webinterface. The following chapter provides an overview over the available parameters and tells you where to find them in the configuration files as well as in the webinterface.

4.1 Configuration Parameter Quick Reference

The following pages will list all parameters available in the configuration files. You will find a short description for every parameter together with it's default value and the location in the webinterface. In some cases you will also find a reference to the sections of this manual, where the usage or the possible values of the parameter are described in detail.

General Settings

These settings configure the general behavior of FSI Cache. They can be found in the "application" and "cache" sections in the main settings-file: *WEB-INF/config/settings.xml*.

Application enabled	
Description	Enables or disables the application
Syntax	Boolean
Default	true
Context	application attribute

Enables or disables the application. This allows disabling the FSI Cache for example for maintenance purposes. If FSI Cache is disabled, requests directed at the application will still be accepted. The client will receive an HTTP status code 503 (Service unavailable) as a response.

standalone	
Description	Enables or disables the connection to FSI Administrator
Syntax	Boolean
Default	false
Context	application element

Allows or disallows an instance of FSI Administrator to connect to this application. Without the connection to FSI Administrator the webinterface cannot be used.

rootredirecturl	
Description	Defines a URL to that the client will be redirected if the servlet request contains no parameters
Syntax	URL
Default	-
Context	application element

If the request is directed to the servlet root and contains no further parameters then the client will receive a redirect header to the specified URL.

rootmode	
Description	Enables or disables root mode
Syntax	Boolean
Default	false
Context	application element

Enable this if FSI Cache is running as the root servlet in the servlet container.

browsercache	
Description	Enables or disables a header allowing clients to cache contents.
Syntax	Boolean
Default	true
Context	application element

If disabled, FSI Cache will include HTTP headers in every response that will forbid the client to cache the received data. The client will be forced to reload the data every time the user visits the appropriate page. Depending on your setup and where your images are used, disabling this will have a noticeable impact on your servers traffic.

servertokens	
Description	Defines aset of headers used in HTTP responses.
Syntax	String – none debug production full
Default	production
Context	application element

FSI Cache will include an HTTP-header in every response sent out. This header identifies the application and may include further information. Turning off this header (setting to "none") is only possible in the enterprise edition of FSI Cache.

precisetypeheaders	
Description	Enables or disables internal specification of the content type.
Syntax	Boolean
Default	true
Context	application element

Enables or disables the internal mechanisms of FSI Cache to determine the content type of the response data. By default FSI Cache will not include a Content-Type header in its HTTP responses. Depending on its configuration the application server may do so, but this is usually not very precise as application servers usually base the decision on the filename suffix. If this option is enabled, FSI Cache will analyze the data and then send a Content Type header. This will slightly reduce the overall application performance, but might be necessary in some environments.

nativequerysupport (Enterprise Edition only)	
Description	Enables or disables accepting requests in native query format.
Syntax	Boolean
Default	false
Context	application element

If enabled, FSI Cache can be addressed using the same request format as the as the FSI Server or the eRez Imaging Server. This for example allows transparent replacement of an Imaging Server with a combination of FSI Cache and any supported Imaging Server.

Statistics	
Description	Enables or disables statistics
Syntax	Boolean
Default	true
Context	application element

Enable or disables FSI Caches statistics collection features.

StatisticsCleanUp	
Description	Defines the time the daily statistics will be aggregated.
Syntax	Time
Default	2:30
Context	application element

Statistics are collected in a database which requires cleaning up once a day. This parameter defines when to start the clean up.

Cache enabled	
Description	Enables or disables caching functionality
Syntax	Boolean
Default	true
Context	cache attribute

This allows disabling the actual caching procedure, for example for setup or testing purposes. FSI Cache will still accept requests and will respond to them as usual, but the content received from the imaging server will no longer be stored in the local cache.

Cache Control Settings

controldaemon	
Description	Enables or disables cache control
Syntax	Boolean
Default	true
Context	cache element

Enables or disables the cache control. If enabled the contents of the cache is scanned in regular intervals. Using the results of this scan the cache will be cleaned according to the other parameters in this section. For example files could be deleted if they are too old or if there are too many files in the cache.

controlday, controlhour, controlminute	
Description	Specifies a fixed time for cache scanning
Syntax	Integer
Default	-1
Context	cache element

Defines exact times when to scan the cache. This way you can time cache scans to take place in low traffic times, e.g. at night or at weekends. Please note that especially with very large amounts of cached files a cache scan can take several hours.

maxcheckpoolsize	
Description	Defines the maximum number of files that can be deleted after one cache scan
Syntax	Integer
Default	2000
Context	cache element

As described above this pool is used to keep track of the oldest files in the cache. (see → 4.5 *Cache Control Service* for details)

maxfileage	
Description	Defines how old a file may be before it is deleted from the cache
Syntax	Timer
Default	30 days
Context	cache element

All files exceeding this defined age will be deleted from the cache. This is not the best way to keep your cache clean as there is no need to delete a file from the cache, if it hasn't changed on the imaging server, no matter how old it is. Use this option only if the Remote Sync (see → 4.4 *Remote Sync Service* for details) feature is not available. Reducing this setting will increase the load on your imaging server.

maxfiles	
Description	Defines the maximum number of files that may be stored in the cache
Syntax	Integer
Default	1000000
Context	cache element

Please note that this is not a hard limit. The cache will be cleaned to stay within the given limit after each scan. In-between scans the limit can be exceeded though.

minfreepartitionspace	
Description	The minimum number of bytes that are to be kept free on the partition used to store the cached files.
Syntax	String – Integer + MB GB
Default	2GB
Context	cache element

This is also not a hard limit. In between cache scans the free partition space can fall below the given value.

poolstrategy	
Description	Regulates the scanning speed.
Syntax	String – slow moderate fast
Default	moderate
Context	cache element

Defines how fast the cache will be scanned. Fast scanning requires more system resources and might have an impact on the performance, especially single core hardware

Remote Synch Settings

These settings configure the behavior of the Remote Sync feature of FSI Cache. They can be found in the "sync" section in the main settings-file: WEB-INF/config/settings.xml. If you are using the webinterface the appropriate menu entry is "Remote Sync".

Sync enabled	
Description	Enables or disabled the Remote Sync Service
Syntax	Boolean
Default	true
Context	synch attribute

Enables or disables the Remote Sync Service. For details on the Remote Sync Service see → [4.4 Remote Sync Service](#) for details.

threads	
Description	Number of threads used to query the imaging server
Syntax	Integer
Default	3
Context	synch element

Defines the number of threads used to query the imaging server. This does not define the maximum number of concurrent requests that are sent to the imaging server, as the Remote Sync requests are sent out during normal operation.

delay	
Description	Time to collect requests
Syntax	Time
Default	20 seconds
Context	synch element

Defines the timespan in which unique requests are collected before they are sent to the imaging server.

poolsize	
Description	The maximum number of unique requests to collect
Syntax	Integer
Default	10
Context	synch element

Defines the number of unique requests that can be collected in one timespan.

DoS Protection Settings

These settings configure protection against Denial of Service attacks. This feature is available in the Enterprise Edition only. Please refer to chapter → 4.6 *DoS Protection (Enterprise Edition only)* for details.

dosprotection enabled	
Description	Enables or disables the DoS protection
Syntax	Boolean
Default	true
Context	protection attribute

Enables or disables the DoS Protection.

ipdos	
Description	Enables or disables the protection against DoS attacks
Syntax	Boolean
Default	true
Context	protection attribute

Enables or disables the protection against Denial of Service attacks. This allows to detect and block too many requests originating from the same source.

ipdossensecount	
Description	The number of allowed requests in the given timespan
Syntax	Integer
Default	1000
Context	protection attribute

Defines the number of allowed requests in the timespan defined by `ipdossensetime`. If this number is exceeded all further requests originating from the same IP address will be blocked.

ipdossensetime	
Description	Timespan in which requests are counted
Syntax	Time
Default	30 seconds
Context	protection attribute

Defines the period of time in which the maximum number of allowed requests are counted.

requestdos	
Description	Enables or disables the protection against DDoS attacks
Syntax	Boolean
Default	true
Context	protection attribute

Enables or disables the protection against Distributed Denial of Service attacks. This allows to detect and block too many requests to the same file originating from different sources.

requestdossensecount	
Description	The number of allowed requests in the given timespan
Syntax	Integer
Default	1000
Context	protection attribute

Defines the number of allowed requests in the timespan defined by requestdossensetime. If this number is exceeded all further requests to the same file will be blocked.

requestdossensetime	
Description	Timespan in which requests are counted
Syntax	Time
Default	30 seconds
Context	protection attribute

Defines the period of time in which the maximum number of allowed requests are counted.

whitelisthosts	
Description	Enables or disabled the usage of a whitelist for allowed IP addresses
Syntax	Boolean
Default	false
Context	protection attribute

If enabled, all requests originating from an IP address contained in the whitelist will be accepted.

blacklisthosts	
Description	Enables or disabled the usage of a blacklist for unwanted referrers
Syntax	Boolean
Default	false
Context	protection attribute

If enabled, all requests containing a referrer matching an entry in the blacklist will be blocked.

blacklistuseragents	
Description	Enables or disabled the usage of a blacklist for unwanted user agents
Syntax	Boolean
Default	false
Context	protection attribute

If enabled, all requests being identified as coming from a blacklisted user agent will be blocked.

Connection Settings

maxIncomingRequests	
Description	Defines the maximum number of requests that are handled simultaneously.
Syntax	Integer
Default	1000
Context	connection element

This allows limiting the number of requests that are handled simultaneously.

maxOutgoingConnections	
Description	Sets the maximum number of simultaneous connections to the imaging server.
Syntax	Integer
Default	800
Context	connection element

Defines the maximum number of connections the server is allowed to keep open to the imaging server(s).

maxTotalConnections	
Description	Sets the maximum number of total connections allowed simultaneously.
Syntax	Integer
Default	800
Context	connection element

Defines the maximum number of connections the server is allowed to keep open.

timeoutSocket	
Description	Defines the socket timeout
Syntax	Integer
Default	20000
Context	connection element

The number of milliseconds to wait when obtaining a socket for a new connection.

timeoutConnection	
Description	The number of milliseconds before a connection times out
Syntax	Integer
Default	12000
Context	connection element

The number of milliseconds to wait before a stale connection is automatically closed.

timeoutIdle	
Description	The number of milliseconds until idle connections are closed
Syntax	Integer
Default	2000
Context	connection element

Defines the number of milliseconds to wait before idle connections are automatically closed.

keepalive	
Description	The keep alive value sent in HTTP headers
Syntax	Integer
Default	3000
Context	connection element

Defines the keep alive value used in HTTP headers.

redirects	
Description	The number of redirects to follow
Syntax	Integer
Default	3
Context	connection element

The maximum number of redirects to follow when requesting data from the imaging server.

retries	
Description	Maximum number of retries
Syntax	Integer
Default	2
Context	connection element

The maximum number of retries before giving up on connecting to the imaging server.

4.2 Profile Settings

These settings are valid for the profile they are defined in. Some of them override the default behaviour described in the section above. These settings are defined in the profile-file located in the WEB-INF/profiles directory of your FSI Cache installation. The name of the file is the name of the profile followed by a .xml suffix.

type	
Description	Defines the profile type
Syntax	String – generic passthrough eRez
Default	generic

Defines the type of profile. This is relevant for the way the underlying imaging server is addressed. Passthrough profiles are only available in the Enterprise Edition.

addresses	
Description	The addresses of the underlying imaging server(s)
Syntax	String
Default	-
Webinterface	Profiles / Basic Profile Settings

The hostname, optional port and priority of the imaging servers that will be used by this profile.

Example: "host.domain.tld:80[15]" where 80 is the port and the value in square brackets is the balancing priority. For details please see section 4.8. The FSI Servers IP address can alternatively be used instead of the hostname. If an IPv6 address is used, this needs to be enclosed in square brackets (IPv6-URL notation).

serverpath	
Description	The path leading to the imaging server
Syntax	String
Default	-

Defines the path leading to the imaging server software.

addressRequired	
Description	Defines if the imaging server address needs to be part of the request
Syntax	Boolean
Default	false

If enabled all requests directed to this profile must contain the address of the imaging server.

forcessl	
Description	Use SSL for communication with imaging server
Syntax	Boolean
Default	false

States if SSL should be used when sending requests to the imaging server.

wraperrors	
Description	Wrap Errors in XML Documents
Syntax	Boolean
Default	false

If enabled error responses are packed in XML Documents and sent out using an HTTP status code of 207. This is primarily used by FSI Viewer instances on the client side.

balancemode (Enterprise Edition only)	
Description	Defines the balancemode for this profile
Syntax	String – none random randomcount loop loopcount
Default	none

Sets the mode used for imaging server load balancing. Please → [4.8 Balancing Imaging Servers \(Enterprise Edition only\)](#) for details on balancing.

alias	
Description	Defines an alias for this profile
Syntax	String
Default	-

If an alias is defined this can be used to address the profile instead of the profile name.

truncateprefixpaths	
Description	The number of path elements to truncate
Syntax	Integer
Default	0

Defines the number of path elements that will be removed from the front of the image path, before this is sent to the imaging server.

securekey	
Description	The key used to decrypt the request path
Syntax	String
Default	-

If this key is set then the received request is expected to be encrypted. For more details on using encryption please see → [7.4 Path encryption](#) for details.

cryptedpaths	
Description	The number of path elements that are encrypted
Syntax	Integer
Default	0

Defines the number of path elements that are expected to be encrypted. See section → [7.4 Path encryption](#) for details.

plainendgap	
Description	Path elements at the end, that are not encrypted
Syntax	Integer
Default	0

Defines the number of path elements at the end of the request that are considered to be not encrypted. See section → *7.4 Path encryption* for details.

forcequality	
Description	Use fixed image quality
Syntax	Integer
Default	-
Webinterface	Profiles / Image Settings

If set, this value will be used as parameter for the image quality in all requests sent to the imaging server by this profile. Any quality parameters contained in the request received by FSI Cache will be ignored.

minquality	
Description	Minimum allowed quality
Syntax	Integer
Default	-

If set, this is the minimum value allowed to use as quality parameter. If a request is received containing a quality parameter below the defined minimum, then the quality parameter will be increased to match this.

maxquality	
Description	Maximum allowed quality
Syntax	Integer
Default	-

If set, this is the maximum value allowed to use as quality parameter. If a request is received containing a quality parameter above the defined maximum, then the quality parameter will be increased to match this.

maxwidth	
Description	Maximum allowed image width
Syntax	Integer
Default	2000

Sets the maximum image width that FSI Cache will deliver. If a larger image is requested the response will contain an error message instead of the image.

maxheight	
Description	Maximum allowed image height
Syntax	Integer
Default	2000

Sets the maximum image height that FSI Cache will deliver. If a larger image is requested the response will contain an error message instead of the image.

fallbackimagepath	
Description	Path to the fallback image
Syntax	String
Default	-

Sets the path to the fallback image in the imaging server. If defined, this image will be delivered to the user if the requested image was not found on the imaging server.

noremotesync	
Description	Disables RemoteSync for this profile
Syntax	Boolean
Default	false

If enabled, the Remote Sync Service will not be used for this profile. This might be useful if this profile addresses an eRez imaging server Version 3. See section → *Remote Sync Service* for details on the Remote Sync Service.

Additionally all settings concerning the Cache Control Service can be made for each profile. A Cache Control Settings made in a profile will always override the appropriate default setting.

4.3 Basic Configuration

The FSI Cache default settings were chosen to fit most scenarios but before using FSI Cache it is necessary to edit or create profiles to match your needs.

Example profiles

FSI Cache comes with three example profiles that only need little adapting before they can be used. The example profiles are located in the profiles directory beneath the WEB-INF directory of the running FSI Cache instance. In order to use one of these profiles rename the file omitting the .dist suffix. Once renamed, all you need to do is fill in the hostname of your imaging server and the profile is usable.

Creating and editing profiles

You can edit the profiles by manually editing the appropriate configuration files. These are located in a directory called "profiles" beneath your WebApplications WEB-INF directory. The name of the file (without the suffix) is also the name of the profile and therefore part of the URL used to access data from this profile. To create a new profile simply copy one of the existing and edit the new file according to your needs. Using the webinterface to edit and create profiles offers a convenient way input mask, including descriptions and help texts for each input field. Click on the "Profiles" entry in the main menu to get a list of all the currently defined profiles. Click on the a profiles "Edit" button to change a profile. To create a new profile use the "Add Profile" Button at the end of the page or copy an existing profile by using a profiles "Copy" button.

4.4 Remote Sync Service

The Remote Sync Service is one of the built-in mechanisms to prevent storing invalid data in the cache. All requests directed at FSI Cache that could be handled using cached files are collected over a predefined period of time. They are then filtered and sent to the imaging server using the HTTP head method. The Last-Modified header in the imaging servers response is analyzed and the service then decides if the cached data is still valid. Invalid data is deleted from the cache immediately. The Remote Sync Service therefore depends on the imaging servers correct handling of the Last-Modified header. As especially older imaging servers do not include correct Last-Modified headers in their responses it is possible to deactivate this service on a per profile basis. In these cases please adapt your cache control service settings for the appropriate profile to compensate for the missing Remote Sync Service.

The optimal settings for the Remote Sync Service highly depend on the usage of your imaging server and cache. Finding the optimal balance between the timespan in which requests are collected the size of the buffer in which these requests are stored is the main task in fine tuning the Remote Sync Service. Choosing a to short

period may result in higher load on the imaging server as the response data for recurring requests will be validated more often. A longer timespan on the other hand might require a larger buffer to hold all requests received, which then will use more local memory resources. As recurring identical requests are only stored in the buffer once, environments with only a few images that are requested often have less problems in finding to optimal settings than environments with requests evenly spread over a lot of images.

The third Remote Sync setting defines the number of threads that are used to send Remote Sync queries to the imaging server. This parameter should be set according to the capacities of the imaging server, but users should keep in mind, that the Remote Sync Service operates independently from the normal service. There will most likely be concurrent requests to the imaging server due to normal operation.

4.5 Cache Control Service

The Cache Control Service is the main service controlling the amount of data cached. It scans the complete cache in regular intervals, amongst others creating a list of the oldest files in the cache. The scan results are then analyzed and if necessary files are deleted according to predefined settings.

As different profiles may require different settings for the Cache Control Service it is not only possible to define default settings for all profiles but also to specifically override one or more settings on a per profile basis.

The Cache Control Service can be configured to ensure that the amount of cached files, the minimum available disk space and the maximum age of the cached content stay within predefined limits. The Cache Control Service will delete as many files as necessary for the cache to stay within these limits after each scan. Although a tolerance margin is included to compensate for the period to the next scan, it is not guaranteed that the cache stays within the defined limits in-between scans.

To limit memory usage of the Cache Control Service the size of the pool to keep track of the oldest files is limited by the configuration. This implicates that the number of files that can be deleted after one scan is also limited. If the more files need to be deleted in order to reach the predefined conditions then an immediate new scan will be triggered.

Cache scanning may be resource consuming, so if possible it should be timed to take place in hours of low traffic. Furthermore the scanning speed has an enormous impact on system performance. Fast scanning can result in very high disk i/o load and can therefore interfere with and slow down content delivery. Slow scanning on the other hand can extend the overall scanning time immensely especially with large caches.

4.6 DoS Protection (Enterprise Edition only)

The Enterprise Edition of FSI Cache is equipped with a request filter that helps preventing Denial-of-Service attacks. On one hand this filter makes use of blacklists to block requests matching predefined criteria, on the other it can block request from certain hosts or to a certain file if they exceed an allowed maximum.

The blacklists allow blocking requests from predefined IP addresses, from specific user agents or such containing unwanted referrer values. Blocking requests from specific IP addresses or address ranges allows the user to effectively prevent content delivery to those addresses. Blacklisting user agents does not enhance security, it can make it harder for someone to mirror your site though. Blocking referrers is a quite effective method to prevent others from directly including images from your server into their websites or using your product pictures in online auctions.

The whitelist allows entering IP addresses and ranges will never be blocked. All requests coming from IP addresses on this list are excluded from all other DoS Protection checks. The requests will be answered normally under all circumstances, as if the request filter was switched off.

The most effective method to prevent Denial-of-Service and Distributed-Denial-of-Service attacks is to stop responding to requests if there are too many coming from the same source or if too many identical requests are received from different sources. In order to achieve this, the request filter analyses all incoming requests. The amount of identical requests or those coming from the same source is counted over a predefined period of time. If the amount exceeds the allowed maximum then any further requests of this type will not be answered. The blocked IP address or request URL will be unblocked as soon as there are no new requests in the given timespan.

4.7 Fallback Imaging Server

FSI Cache can be configured to use more than one imaging server in a single profile. If a second server address is specified then this will automatically be used as the backup address. If FSI Cache notices that the primary imaging server is no longer reachable, it will switch to the backup server until the primary server is back on line. In order for the fallback mechanism to operate as expected it is necessary to ensure that the images provided are identical on all imaging servers.

If you want FSI Cache to actively balance requests between two or more imaging servers, then this is possible with the enterprise edition of FSI Cache, please see the following section for details.

4.8 Balancing Imaging Servers (Enterprise Edition only)

The Enterprise Edition of FSI Cache supports various methods of balancing requests between two or more imaging servers. All methods have in common that they will temporarily remove an imaging server from their list, if it is not reachable and continue normal operation with the remaining servers.

There are two main balancer modes, loop and random. As the names imply, the loop balancer mode directs a request to the server in the list that is after the one the last request was sent to, whereas the random balancer mode randomly chooses a server from the list. Additionally both modes support prioritizing the imaging servers to support more and less powerful imaging servers. Choose the prioritized version of the desired mode in the balancer mode dropdown and add priorities to the server entries in the addresses section of the webinterface.

Please note that is essential for all imaging servers involved in the balancing that the images provided by the servers are identical. Especially when the Remote Sync Service is used it is absolutely necessary that the file modification dates of the images on the imaging serves do not differ. If you are using FSI Cache in a high traffic environment and need to use more than one instance of FSI Cache, please refer to chapter → *6.1 Balanced FSI Cache Servers*.

4.9 Integrating FSI Cache into your IT environment

This chapter describes some issues of integrating FSI Cache into your IT environment and how FSI Cache interacts with other software.

Logging

Like all other Java based FSI Applications, FSI Cache uses Log4J as a logging framework. This allows you to configure log level, file format and output file to your needs, providing the opportunity to integrate the log output into your log monitoring and management applications. FSI Cache uses two different logs: the application logs contain events and errors from within the application and the request log, which contains all requests that FSI Cache sends out to the imaging server. By default the output of both logs will end up the application servers log directory in log files that will be rotated on an daily basis. The log configuration files can be found in the config directory beneath the WEB-INF directory of the running

FSI Cache instance. For more information on the format of the log configuration files and the configuration options see the Log4j website at [<http://logging.apache.org/log4j>].

Sharing the application server

Strictly speaking there is no reason why the application server running FSI Cache should not also run other applications simultaneously. Depending on the utilization of FSI Cache though, please consider that a caching system might consume a lot of disk space and also a lot of bandwidth during peak times.

5 Configuration using Administrator

5.1 Connecting to your FSI Cache from FSI Administrator

After logging in to your FSI Administrator servlet, choose "Add Software" from the My Software" Menu. The Connect Software form is displayed. Please choose "FSI Server" from the application dropdown, and enter the full qualified domain name,

the port the application is listening to and the application path. You may choose to add a short description to help identify the server. If the connection to FSI Server was successfully, the server is listed in the "My Software" overview.

If the connection to FSI Server was successfully, the server is listed in the "My Software" overview.

5.2 Administrating FSI Cache

After connecting to your FSI Server, you will see the Administration interface with the following entries in the main application menu.

Profiles

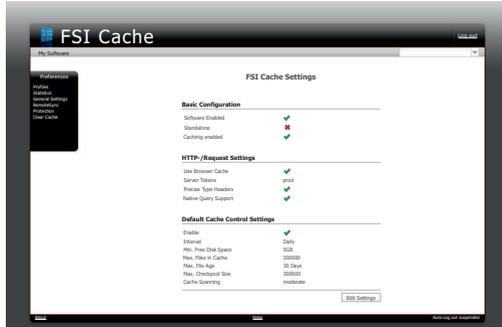
This screen will display an overview of the defined cache profiles.

Statistics

Here you can see request and traffic statistics of FSI Cache. You can also see Fallback image and erroneous requests if any occurred.

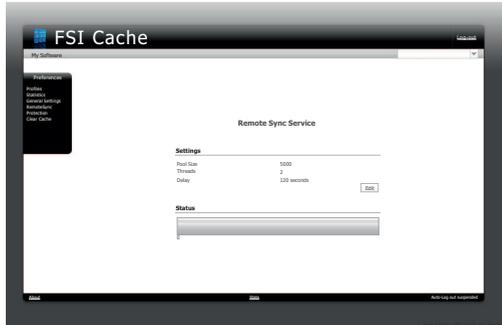
General Settings

This lets you enable or disable the software, configure whether the server is running standalone and define HTTP-/Request settings. You can also define the Cache Control Settings, which define the max. age of files that have been cached, the max. amount of files in the cache.



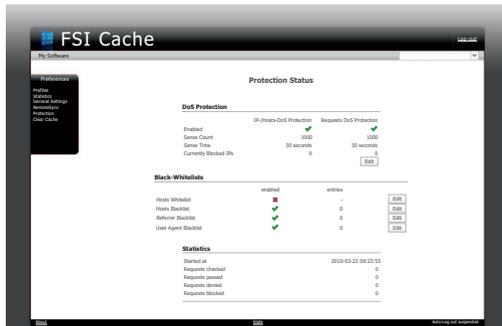
RemoteSync

This automatically retrieves images from the imaging server. This feature is handy for large image pools so the FSI Cache starts caching images independent from User initiated requests. On this screen you can check the current status of the remotesync service.



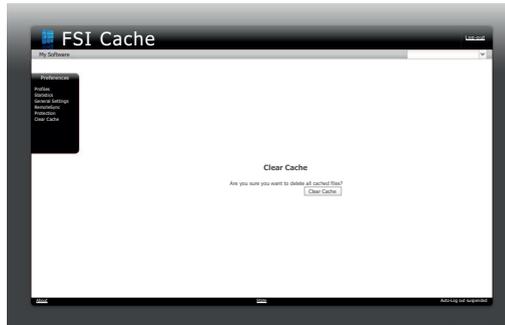
Protection

FSI Cache features DoS protection and blacklist/ or whitelists. The DoS protection denies requests from a certain client. If the amount defined by "Sense Count" is exceeded within the "Sense Time". A blacklist contains a list of Clients that are blocked from accessing FSI Cache, while client requests off the whitelist are processed.



Clear Cache

You can clear the entire cache on this screen.



6 Advanced Setups

6.1 Balanced FSI Cache Servers

Balancing two or more FSI Cache Servers is a common practice and is a useful way to cope with increasing demand. If you are planning to use a hardware load balancer or simply setup DNS based balancing, it is important to ensure that the underlying imaging servers deliver correct Last-Modified headers. If the Last-Modified headers delivered from the imaging server are not correct, the Last-Modified headers included in responses from the FSI Cache Servers will also differ, resulting in an unwanted behavior on behalf of the balancer and/or any involved proxies and clients.

6.2 Cascaded FSI Cache instances

Under certain circumstances it might be useful to cascade FSI Cache servers. For example if you have a lot of FSI Cache instances in a balancer array to cope with the traffic, then changes of frequently requested images might cause a very high load on the imaging server, as it would have to process the same image for all cache servers in a short period of time. In a scenario like this an FSI Cache in front of the imaging server would be useful. This further FSI Cache would have to be an Enterprise Edition as it would have to be able to accept eRez style queries. See the next section for more example setups.

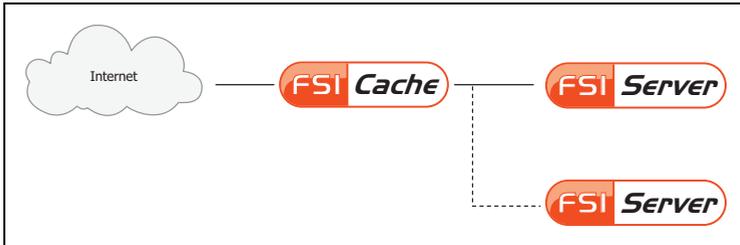
6.3 Example Setups

The following figures show possible setup combinations of FSI Cache and imaging servers. Starting from the simplest of all setups to the most powerful, they are only examples to indicate that FSI Cache environments can be scaled to exactly match your needs.

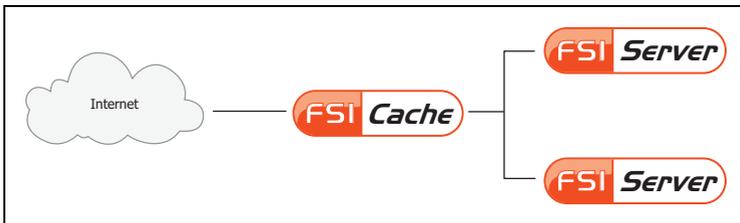
A simple setup with one FSI Cache: Compared to a standalone imaging server this setup offers improved content delivery response times and reduced imaging server load.



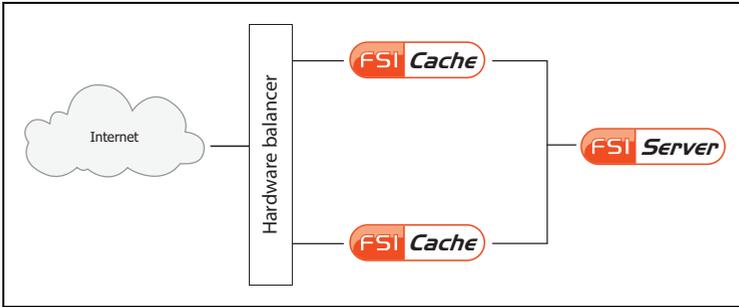
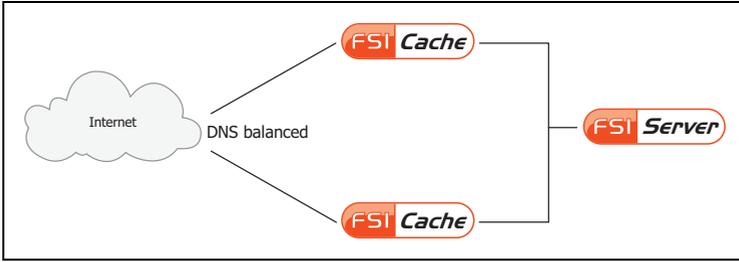
Another fairly simple setup using a fallback imaging server.



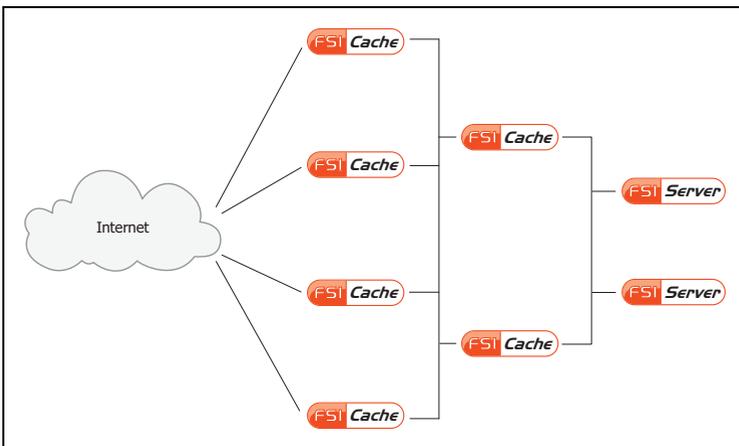
One FSI Cache balancing between two (or more) imaging servers. This setup is suitable for environments with moderate traffic rates but a fluctuating image inventory.



Balanced FSI Cache Servers accessing a single imaging server are perfect for high traffic environments with a fairly static image inventory.



Cascaded Caches with two balanced imaging servers: This setup is the ultimate all-in-one solution, providing as many first level caches as needed to serve the requested bandwidth. Two second level caches reduce the load on the imaging servers and together with the two imaging servers all single points of failure are eliminated.



7 Using FSI Cache

7.1 Addressing

Section 3.3 gave a short introduction on how to address FSI Cache. The exact paths to access content depend on multiple settings. This section will describe all parameters relevant for addressing FSI Cache content.

The following examples will all access the same image on the imaging server. The path to this image on the server being path/to/image.tif.

As already shown the simplest URL to access the example image would be:

```
http://your.fsi-server.com/fsicache/profile/path/to/image.tif/width_height
```

This requires that `addressrequired` is set to `false` and `truncateprefixpaths` is set to `0` in the used profile.

If `addressrequired` is turned on in the profile, then the request URL would have to be extended to the following form:

```
http://your.fsi-server.com/fsicache/profile/imagingserver/path/to/image.tif/width_height
```

If `truncateprefixpaths` is increased to `2` the URL would be:

```
http://.../imagingserver/x/y/path/to/image.tif/width_height
```

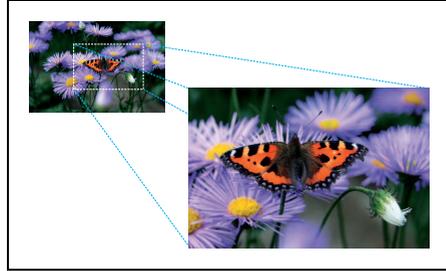
where the actual content of `x` and `y` is not of interest.

7.2 Image Ranges

Regardless of the configuration the URL can be extended to include further commands. Going back to the first example for simplicity the URL can contain an image range:

```
http://.../path/to/image/width_height_0c5_0_0c5_0c5
```

For example would result in an image containing the top right quarter of the original image. The first two parts after the width and height statements define the top left corner of the rectangular range starting from the top left corner of the original image. In this case 50% (0c5) to the right and 0% (0) to the bottom. The next two parts define the size of the range, in this case 50% (0c5) of the original images width and 50% (0c5) of the original images height. The resulting image range will then be scaled to fit the requested width and height. The following figure illustrates the image range selection.



7.3 Further server commands

By further extending the request URL it is possible to send additional parameters to the imaging server, that will be applied when creating the image. These commands are appended to the end of the URL separated by "_fx_". The following example will use the imaging servers built-in rotation function to rotate the image 180°.

```
http://your.fsi- server.com/fsicache/profile/path/to/image.  
tif/width_height_fx_effects%3dflip(vertical)
```

As this is a simple URL all special characters need to be url-encoded, explaining why angle=180 needs to be written as angle%3d180. Multiple server commands can be used in the same request by separating them with an underscore. With the exception of "quality" these commands are not interpreted or checked by FSI Cache.

7.4 Path encryption

In order to prevent people from systematically downloading images, image regions or image sizes it is possible to encrypt parts of the request URL. The webinterface offers encrypting example paths, if you want to use encryption in a large scale environment, please contact NeptuneLabs. You will then receive example code, showing you how to integrate the encryption algorithm into your application or Content Management System.

The number of parts of the URL that are expected to be encrypted is specified by the "cryptedpaths" setting in the profile. By default the parts are counted from the end of the URL. Specifying the "plainendgap" setting allows you the shift the

encrypted parts forward. The webinterface allows modifying these settings and shows an example of what a URL directed to this profile should look like.

8 Known Issues

The following list contains hints that should be considered when using caching systems and especially FSI Cache.

8.1 Limit available disk space

Limit the amount of disk space the cache may use, or in this case setup the Cache Control Daemon to keep a minimum of free disk space.

8.2 Imaging server licenses and templates

If a file changes on the underlying server, but the file modification time does not change, then the cache will not realize the change and the cached data will not be updated.

This is will for example be the case, if you modify the metadata template, or if you license the imaging server. The delivered files will then no longer contain watermarks, but the cache has no way of knowing. In these cases you must clear the cache manually.

8.3 eRez headers

Please do not modify the headers.xml on the eRez imaging server, as this is essential for the correct implementation of the Last-Modified headers in the imaging server responses.

Index

Administrating FSI Cache	33	blacklisthosts	19
Advanced Setups	37	blacklistuseragents	20
Balanced FSI Cache Servers	37	browsercache	12
Cascaded FSI Cache servers	37	Cache enabled	14
Example Setups	37	controldaemon	14
Configuration	11	controlday, controlhour, con-	
Adressing	41	trolminute	14
Balancing Imaging Servers	30	cryptedpaths	24
Basic Configuration	27	delay	17
Cache Control Service	28	dosprotection enabled	17
DoS Protection	29	fallbackimagepath	26
Fallback Imaging Server	29	forcequality	25
Further server commands	42	forcessl	23
Image Ranges	41	ipdos	18
Integrating FSI Cache into your IT		ipdossensecount	18
environment	30	ipdossensetime	18
Path encryption	42	keepalive	21
Profile Settings	22	maxcheckpoolsize	15
Remote Sync Service	27	maxfileage	15
Configuration using Administrator	33	maxfiles	15
Connecting to your FSI Cache from FSI		maxheight	26
Administrator	33	maxIncomingRequests	20
Essentials	9	maxOutgoingConnections	20
Addressing	9	maxquality	25
How it works	9	maxTotalConnections	20
Keep the cache clean	10	maxwidth	26
Profiles	9	minfreepartitionspace	16
FSI Cache		minquality	25
Compatibility	6	nativequerysupport	13
Preface	5	noremotesync	26
System Requirements	5	plainendgap	25
Installation	7	poolsize	17
Platform independent installer	7	poolstrategy	16
Windows Installer	7	precisetypeheaders	13
Known Issues	45	redirects	22
eRez headers	45	requestdos	18
Imaging server licenses and tem-		requestdossensecount	19
plates	45	requestdossensetime	19
Limit available disk space	45	retries	22
Parameter	11	rootmode	12
addresses	22	rootredirecturl	12
addressRequired	23	securekey	24
alias	24	serverpath	23
Application enabled	11	servertokens	12
balancermode	24	standalone	11

Statistics	13
StatisticsCleanUp	14
Sync enabled	16
threads	17
timeoutConnection	21
timeoutIdle	21
timeoutSocket	21
truncateprefixpaths	24
type	22
whitelisthosts	19
wraperrors	23
Using FSI Cache	41

FSI Cache

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.

© 2009-2015 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