



**MedDream DICOM Viewer**

**Servicing MANUAL**

**(version 7.0.0-736e481a2f22)**

© 2018, Softneta UAB, Kaunas

All rights reserved in the event of granting of patents or registration as a utility patent.

All names of companies and products mentioned in this user's manual may be trademarks or registered trademarks. References to products of other manufacturers are for information purposes only. Such references are intended neither as an approval nor a recommendation of these products. Softneta UAB accepts no liability for the performance or use of such products.

Other brand names, software and hardware names used in this user's manual are subject to trademark or patent protection. The quoting of products is for informational purposes only and does not represent a trademark misuse.

This Servicing Manual is protected by copyright. Unless expressly authorized in writing, dissemination, duplication or other commercial exploitation of this documentation set or communication of its contents or parts of it is not permitted. In case of infringement, the violator may be liable to pay compensation for damages.

Specifications due to technical developments are subject to change. This Servicing Manual is not subject to the revision service. Please contact the manufacturer or authorized dealer to request the latest edition of Servicing Manual.

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Minimal server side requirements</b>	<b>1</b>
2.1	Minimal hardware requirements	1
2.2	Minimal memory requirements	1
2.3	Minimal software requirements	1
2.3.1	Supported operating systems	1
2.3.2	Supported WEB servers	2
2.3.3	Supported PHP scripting interpreters	2
<b>3</b>	<b>General considerations</b>	<b>2</b>
<b>4</b>	<b>Apache configuration</b>	<b>3</b>
4.1	Apache modules	3
4.2	AllowOverride directive	3
4.3	Reverse proxy	4
<b>5</b>	<b>Deployment under Windows operating systems</b>	<b>4</b>
<b>6</b>	<b>Deployment under Linux operating systems</b>	<b>5</b>
<b>7</b>	<b>Deploying the Java-based core</b>	<b>7</b>
7.1	Configuration	7
7.2	Running as a service	9
7.2.1	Windows	9
7.2.2	Linux (System-V init)	9
7.2.3	Linux (systemd)	9
7.2.4	Linux (upstart)	10
<b>8</b>	<b>Integration with PACS</b>	<b>10</b>
8.1	Integration modes	10
8.2	Direct access to PACS database mode	10
8.2.1	PacsOne	10
8.2.2	DCM4CHEE	12
8.2.3	ClearCanvas	13
8.2.4	Conquest	13
8.2.5	Aycan	14
8.3	Web Access to DICOM Persistent Objects (WADO) mode	14
8.4	DICOM mode	15
8.5	File system access mode	16
8.6	Plugins for Java-based core	17
<b>9</b>	<b>Image Access from hospital information system (HIS)</b>	<b>22</b>
9.1	Generic HIS integration	22
9.2	Specification of thumbnails service	24
9.3	Specification of Live Stream	25
9.4	Further reading	25
<b>10</b>	<b>Additional software</b>	<b>26</b>
10.1	3D service	26
10.1.1	3D server configuration	26
10.1.2	Network settings (ports and SSL)	26
10.1.3	Quality / Performance	26
10.2	Browser plugin	27
10.3	OpenAM verification service	27

<b>11 MedDream and OAuth</b>	<b>28</b>
11.1 Installation . . . . .	28
<b>12 MedDream and external proxies</b>	<b>29</b>
<b>13 Security considerations</b>	<b>29</b>
13.1 Search engines . . . . .	29
13.2 The subdirectory 'log' . . . . .	30
13.3 The subdirectory 'temp' . . . . .	30
13.4 printVersions.php . . . . .	30
13.5 DCM4CHEE 2.x . . . . .	31
13.6 The subdirectory 'scripts' . . . . .	31
13.7 prepared.php . . . . .	31
13.8 Embedding into IFRAME etc . . . . .	31
<b>14 Localization</b>	<b>32</b>
<b>15 Rebranding</b>	<b>32</b>
<b>16 Troubleshooting</b>	<b>37</b>
16.1 Log files of the legacy backend . . . . .	37
16.2 Audit logs . . . . .	37
16.3 Log files of the Java-based core . . . . .	37
16.4 Browser download path selection . . . . .	37
16.4.1 Google Chrome . . . . .	37
16.4.2 Mozilla Firefox . . . . .	38
16.4.3 Microsoft Edge . . . . .	38
16.4.4 Safari . . . . .	38
16.4.5 Internet Explorer 11 . . . . .	38



# 1. Introduction

MedDream DICOM Viewer is a HTML based package for PACS server which is designed to aid professionals in every day's decision making process, connecting all the medical data into a unified and fast performing network. MedDream ensures a fast and reliable way to search, present and analyze the medical data (images and video files) on various devices: computers, smart phones, tablets and so forth.

MedDream covers: radiology, cardiology, oncology, gastroenterology and many other fields of medical application. It seamlessly integrates with various medical imaging devices, such as: ultrasound (US), magnetic resonance (MRI), positron emission tomography (PET), computed tomography (CT), endoscopy (ES), mammography (MG), digital radiography (DR), computed radiography (CR) ophthalmology, and so forth.

Core MedDream DICOM Viewer uses are:

- Replacement of hard copies, e.g. film archives, paper documents, etc.
- Remote access. MedDream provides a possibility to be mobile and work from any place in the world where the Internet is accessible. More than one person can access and view medical records at one time. Such functionality speeds up the collaboration among the professionals. So that a doctor in the hospital and a doctor that is in the different location may view the medical data and discuss about it simultaneously. The patient's medical history, various studies and images are found much faster comparing to the conventional paper-based methods.
- MedDream can be used as a standalone WEB Viewer or integrated into MedDream PACS, PacsOne PACS, dcm4chee Archive, Conquest PACS, ClearCanvas PACS systems. Moreover, MedDream can be adapted to client's PACS system and easily integrated into RIS/HIS workflow.
- MedDream has multiple functions such as search of studies, viewing, analyzing, saving, exporting, forwarding images and videos, etc.

## 2. Minimal server side requirements

### 2.1. Minimal hardware requirements

Parameter	Requirement
Processor	2.33GHz or higher x86-compatible
Memory	4 GB
Hard drive	10 GB (RAID 1, RAID 5, RAID 10)
Network Interface	100 Mbit/s

### 2.2. Minimal memory requirements

Minimal memory requirements for the best performance of the software:

- 8 GB of RAM if you plan to open more than 800 images (CT & MRI, PET-CT).
- 12 GB of RAM for more than 1500 images (multi-slice CT & PET-CT).
- 16 GB of RAM for more than 3000 images (cardiac or functional imaging).

### 2.3. Minimal software requirements

#### 2.3.1. Supported operating systems

MedDream supports following operating systems:

- Windows Server 2008 (32 bit and 64 bit) and newer
- Windows 7 (32 and 64 bit) and newer

- Linux (32 bit and 64 bit, with glibc version  $\geq 2.7$ ).

### 2.3.2. Supported WEB servers

MedDream supports Apache HTTP server. Versions from 2.2 are supported. Under Linux operating systems distribution supplier package shall be used. Under Windows operating systems binary packages from third parties (such as XAMPP) may be used or compiled from source.

### 2.3.3. Supported PHP scripting interpreters

MedDream requires PHP server-side scripting language interpreter integrated into the web server as a SAPI module. PHP versions from 5.3 to 5.6 are supported. Under Linux operating systems, distribution supplier package shall be used. Under Windows operating systems, binary packages from third parties may be used or compiled from source.

Under Windows, a thread-safe (TS) 32-bit PHP build is required. The DLLs included won't work with a 64-bit PHP, like in 64-bit builds of WampServer.

Under Linux, a non-TS PHP build is required. This is related to a more common build of Apache, the "prefork". The "worker" build is incompatible.

PHP 5.3 is built for Windows either with VC6 or VC9 runtime. Take note which version you have, and use a corresponding .dll during installation as explained below. Otherwise you'll get an error message like this one:

```
Module compiled with build ID=API20090626,TS,VC6
PHP compiled with build ID=API20090626,TS,VC9
These options need to match
```

The default value for `memory_limit` (php.ini) is often 128M or a similar small value. Large images like CR, DX and multiframe US might require a lot more. For 32-bit Apache and 2 GB of physical memory, you can start from 1536M and experiment further, as values too close to 2 GB might cause poor handling of "out of memory" situations.

MedDream will inform about missing PHP extensions that are critical to its operation. However you can try to install in advance those not listed by `phpinfo()` (or, in some cases, by `php -m`):

- required: `xml`, `SimpleXML`, `json`, `curl` and, in case of *Direct access to PACS database mode*, a corresponding database driver (`mysqli`, `pgsql`, etc)
- recommended: `mcrypt`

## 3. General considerations

During an upgrade it's recommended to use the included `config.sample-*.php` as a template and then update the resulting `config.php` according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

After major changes in `config.php` (especially `$pacs` parameter) it might be required to restart the Java-based core; see *Deploying the Java-based core*.

MySQL from XAMPP 1.8 for Windows: if the web interface connects to the database very slowly (each attempt takes about 1 second), then try adding `bind-address = ::` to `my.ini`.

While still in demo mode, you **must** remove the existing `meddream.lic` after changing computer's host name (any OS), reinstalling the OS (Windows) or changing the MAC of any network card (Linux). A correct file will be created automatically. Otherwise a commercial license generated from your `meddream.lic` will not work. This also means that Windows must be activated in advance!

Under Linux, the .so file requires glibc library version 2.7 or later.

Starting from 5.5, support for DICOM overlays is on by default and MedDream will attempt to patch the image with data from the (6000,3000) tag. If some images fail to open after upgrade to 5.5 and logs contain the

string “Error” related to output of `meddream_thumbnail / meddream_convert2`, then try to turn off the overlays by adding `meddream.overlays_enabled = 0` to `php.ini`.

Starting from 6.0, Java 8 is needed regardless of the PACS as parsing of some image types is implemented in a separate Java application. Later versions are increasingly replacing other functionality with Java-based implementation. See [Deploying the Java-based core](#).

The following instruction assumes that the system already meets prerequisites and runs Apache httpd server with PHP server-side scripting interpreter.

## 4. Apache configuration

### 4.1. Apache modules

The following Apache modules must be loaded:

- `mod_deflate`
- `mod_filter`
- `mod_proxy`
- `mod_proxy_http`
- `mod_rewrite`
- `mod_proxy_wstunnel`

Under Windows operating systems, the following lines in the main Apache configuration file must be uncommented or added:

```
LoadModule deflate_module modules/mod_deflate.so
LoadModule filter_module modules/mod_filter.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule proxy_wstunnel_module modules/mod_proxy_wstunnel.so
```

Under Debian based systems one may use the `a2enmod` command:

```
a2enmod deflate
a2enmod filter
a2enmod rewrite.load
a2enmod proxy.load
a2enmod proxy_http
a2enmod proxy_wstunnel
```

For other distributions, consult the distribution documentation.

Note: Apache service restart or reload is required after modifying `httpd.conf` or enabling modules by other methods.

### 4.2. AllowOverride directive

The `<Directory>` clause that defines MedDream installation directory (or some of its parent directories) must allow `All` or at least `FileInfo`:

```
AllowOverride FileInfo
```

NOTE: the [Security considerations](#) chapter additionally suggests `Limit/AuthConfig` and `Options`. When adding them, do not drop `FileInfo`.



### 4.3. Reverse proxy

A reverse proxy must be set up for Java-based core. The following assumes that MedDream opens as / meddream:

```
<IfModule proxy_module>
  <IfModule proxy_http_module>
    ProxyPass /meddream/api http://127.0.0.1:8080/ nocanon
    ProxyPassReverse /meddream/api http://127.0.0.1:8080/ nocanon
    ProxyPass /meddream/api/study/open ws://127.0.0.1:8080/study/open
    ProxyPassReverse /meddream/api/study/open ws://127.0.0.1:8080/study/open

    #RewriteEngine on
    #RewriteCond %{HTTP:UPGRADE} ^WebSocket$ [NC]
    #RewriteCond %{HTTP:CONNECTION} Upgrade$ [NC]
    #RewriteRule /meddream/api/(study/open.*) ws://127.0.0.1:8080/$1 [P]
  </IfModule>
</IfModule>
```

The commented-out part is usually not required and should remain inactive.

If MedDream opens as /, then just remove the part “/meddream” from all lines. Otherwise you might need to change “/meddream” to the actual Web context path.

If you changed the Application TCP/IP port in the `application.properties` file (see *Deploying the Java-based core*), then “8080” above must be updated with the same value.

## 5. Deployment under Windows operating systems

It is assumed that Apache document root is located at `C:\XAMPP\Apache\htdocs`, and PHP is located at `C:\XAMPP\PHP` and user has administrative permissions.

1. Copy meddream folder to `C:\XAMPP\Apache\htdocs`

(the result: `C:\XAMPP\Apache\htdocs\meddream`)

If you are updating an older version of MedDream:

- a) Please back up the old version (so you can go back any time).
- b) Please do not replace files you changed in last version. You must carefully synchronize the new version.

2. (optional) From `C:\XAMPP\Apache\htdocs\meddream` Copy and Replace `applet.php` to `C:\XAMPP\Apache\htdocs`.

It is advised to make a backup beforehand in case you'll need to uninstall MedDream later.

3. From `C:\XAMPP\Apache\htdocs\meddream`

copy `php5.3_meddream-VC6.dll` to `C:\XAMPP\PHP\ext` (if you are using PHP 5.3.x, VC6 build)

copy `php5.3_meddream-VC9.dll` to `C:\XAMPP\PHP\ext` (if you are using PHP 5.3.x, VC9 build)

copy `php5.4_meddream.dll` to `C:\XAMPP\PHP\ext` (if you are using PHP 5.4.x)

copy `php5.5_meddream.dll` to `C:\XAMPP\PHP\ext` (if you are using PHP 5.5.x)

copy `php5.6_meddream.dll` to `C:\XAMPP\PHP\ext` (if you are using PHP 5.6.x)

4. Add to `php.ini` file

`extension=php5.3_meddream-VC6.dll` (if you are using PHP 5.3.x, VC6 build)

`extension=php5.3_meddream-VC9.dll` (if you are using PHP 5.3.x, VC9 build)

`extension=php5.4_meddream.dll` (if you are using PHP 5.4.x)

extension=php5.5\_meddream.dll (if you are using PHP 5.5.x)

extension=php5.6\_meddream.dll (if you are using PHP 5.6.x)

5. Specify date.timezone in php.ini:

```
[Date]
; Defines the default timezone used by the date functions
date.timezone = "America/Chicago"
```

The list of possible values may be obtained from <http://www.php.net/manual/en/timezones.php>.

6. Restart Apache

7. Make sure meddream\bin\deleteTemp.bat is run at least every night by Task Scheduler. See chapter "INSTALLATION" in quick\_install-Scripts.txt for a recipe. It might be needed to specify the full path to php.exe in deleteTemp.bat.

## 6. Deployment under Linux operating systems

Following guide assumes that Apache document root is located at /var/www/html and PHP configuration is located at /etc/php and user has root permissions.

1. Copy the "meddream" directory to /var/www/html so that the result is /var/www/html/meddream.
2. Adjust permissions: no less than

```
# required for the Java Core
chown www-data:www-data -R /var/www/html/services
find /var/www/html/services -type d -exec chmod 775 {} \;
find /var/www/html/services -type f -exec chmod 664 {} \;

# required for Meddream
chown www-data:www-data -R /var/www/html/meddream
find /var/www/html/meddream -type d -exec chmod 775 {} \;
find /var/www/html/meddream -type f -exec chmod 664 {} \;
find /var/www/html/meddream -iname "*.sh" -exec chmod 775 {} \;

# required both for PHP and Java Core
chmod a+x /var/www/html/meddream/dcm4che/bin/*

# required for displaying SR modality files
chmod a+x /var/www/html/meddream/dcm4che/dsr2html
```

3. If you are updating an older version of MedDream:

Please back up the old version (so you can go back any time).

Please do not replace files you changed in last version. You must carefully synchronize the new version

4. From /var/www/html/meddream:

copy php5.3\_meddream.so to /usr/lib/php/modules (if you are using PHP 5.3.x)

copy php5.4\_meddream.so to /usr/lib/php/modules (if you are using PHP 5.4.x)

copy php5.5\_meddream.so to /usr/lib/php/modules (if you are using PHP 5.5.x)

copy php5.6\_meddream.so to /usr/lib/php/modules (if you are using PHP 5.6.x)

copy php5.3\_meddream-x86\_64.so to /usr/lib64/php/modules (if you are using PHP 5.3.x under a 64-bit OS)

copy php5.4\_meddream-x86\_64.so to /usr/lib64/php/modules (if you are using PHP 5.4.x under a 64-bit OS)

copy `php5.5_meddream-x86_64.so` to `/usr/lib64/php/modules` (if you are using PHP 5.5.x under a 64-bit OS)

copy `php5.6_meddream-x86_64.so` to `/usr/lib64/php/modules` (if you are using PHP 5.6.x under a 64-bit OS)

#### 5. Add to `php.ini` file

`extension=php5.3_meddream.so` (if you are using PHP 5.3.x)

`extension=php5.4_meddream.so` (if you are using PHP 5.4.x)

`extension=php5.5_meddream.so` (if you are using PHP 5.5.x)

`extension=php5.6_meddream.so` (if you are using PHP 5.6.x)

`extension=php5.3_meddream-x86_64.so` (if you are using PHP 5.3.x under a 64-bit OS)

`extension=php5.4_meddream-x86_64.so` (if you are using PHP 5.4.x under a 64-bit OS)

`extension=php5.5_meddream-x86_64.so` (if you are using PHP 5.5.x under a 64-bit OS)

`extension=php5.6_meddream-x86_64.so` (if you are using PHP 5.6.x under a 64-bit OS)

Depending from distribution PHP configuration file may be stored in different location. Please consult distribution documentation.

#### 6. Specify `date.timezone` in `php.ini`:

```
[Date]
; Defines the default timezone used by the date functions
date.timezone = "America/Chicago"
```

The list of possible values may be obtained from <http://www.php.net/manual/en/timezones.php>.

#### 7. Restart Apache

8. Make sure `meddream/bin/deleteTemp.sh` is run at least every night by cron. See chapter “INSTALLATION” in `quick_install-Scripts.txt` for a recipe. Furthermore, check if MedDream can still write to its log files after this script has been run.

9. `mkisofs` is required for the `Export` command. If your distribution does not provide `mkisofs`, `genisoimage` package may be used. In that case a symlink to `genisoimage` is required:

```
ln -s /usr/bin/genisoimage /usr/bin/mkisofs
```

10. To display SR files, we are using a prebuilt binary `meddream/dcmtd/dsr2html` from DCMTK 3.6.3. If your distribution provides another version of it that works better and still understands command line options passed by `SR.php`, then you’ll need to replace `dsr2html` and `dicom.dic` with symlinks to your copies. Also please note this is a 32-bit binary, therefore some 64-bit systems might need an additional 32-bit version of `glibc` (for example, `glibc.i386`).

11. `ffmpeg` is required in order to display thumbnails of MPEG2 DICOM files and non-BD-compatible MPEG4 videos (Transfer Syntax UID 1.2.840.10008.1.2.4.102). If your distribution does not provide `ffmpeg`, `avconv` package may be used. In that case symlink to `avconv` binary is required:

```
ln -s /usr/bin/avconv /usr/bin/ffmpeg
```

12. To display videos of any kind, MedDream v5.5+ needs `FFmpeg` dated not older than 2014-03-08. If yours is older and can’t be updated, then you need at least to switch to the legacy behavior by changing `const USE_FFMPEG_SUBFILE = true` in `flv.php` to an opposite value (false).

13. To display non-BD-compatible MPEG4 videos (Transfer Syntax UID 1.2.840.10008.1.2.4.102), latest version of `FFmpeg` might be required.

14. Starting from 5.5, MedDream adds the option “-y” to `FFmpeg` command line when extracting a thumbnail from a video file. If you are forced to use an older version of `FFmpeg` that doesn’t understand this option, then add the following to `php.ini`:

```
meddream.thumb_ffmpeg_cmdline = "-i ?I -f image2 ?O"
```

## 7. Deploying the Java-based core

### 7.1. Configuration

The Java application can be found in the folder “services” of the MedDream installation archive.

For security considerations please **move this folder to a location that isn’t accessible from Web**. Particularly, the file `application.properties` usually contains login credentials and log files created below this directory might contain sensitive information.

1. Create a file `application.properties` in this folder (you can use the sample bundled with MedDream) and add the following configuration parameters:

1. Location of the legacy endpoint “Routes.php”

```
com.softneta.meddream.url = MEDDREAM_URL/Routes.php
```

Java-based core calls legacy backend for metadata etc.

**Example:** `com.softneta.meddream.url=http://127.0.0.1/Routes.php`

2. Location (directory only) of `meddream.lic`

```
com.softneta.license.licenseFileLocation = PATH_TO_MEDDREAM_LIC_DIRECTORY
```

The HTML Viewer currently uses the Java-based core for the Register function that updates the file `meddream.lic`.

Examples:

```
com.softneta.license.licenseFileLocation = ../meddream
com.softneta.license.licenseFileLocation = /var/www/html
com.softneta.license.licenseFileLocation =
C:\\MedDreamPACS\\MedDream
```

(WARNING: If you need to use Windows paths with backslashes, then every backslash must be doubled.)

3. Remote connection string and local AE Title for saving annotations and key objects

```
com.softneta.meddream.dcmsnd.connect = REMOTE_CONNECTION_STRING
com.softneta.meddream.dcmsnd.bind = LOCAL_AE_TITLE
```

Created DICOM objects will be sent to the PACS over DICOM C-STORE. The PACS is specified by a connection string in form `AE_TITLE@HOST_OR_IP:PORT`. The local AET identifies MedDream. Example:

```
com.softneta.meddream.dcmsnd.connect = DCM4CHEE@192.168.1.
200:11112
com.softneta.meddream.dcmsnd.bind = MEDDREAM
```

NOTE: in a rare event when the PACS requires client traffic to come from a certain source port, `com.softneta.meddream.dcmsnd.bind` can include hostname/IP and port in the same fashion.

4. For linux. FFmpeg installation required for video thumbnail generation.

```
com.softneta.thumbnails.ffMpegExecutable= PATH_TO_FFmpeg
```

5. Application TCP/IP port (optional)

```
server.port = APPLICATION_PORT
```

Allows to choose a different port if the default one, 8080, is occupied. Example: `server.port=8090`.

In this case you'll also need to update the port part of `$core_addr` (`config.php`) to the same value.

## 6. Path to MedDream temp. files

```
com.softneta.meddream.tempDir = PATH_TO_MEDDREAM_TEMP_FILES
```

Allows to specify the path to location where MedDream temp files are stored. Example: `../meddream/temp, /tmp/meddream, E:\mdtmp`.

NOTE: in 7.0 some cached files might need paths longer than 260 characters which are not allowed. The only solution is to have `com.softneta.meddream.tempDir` as short as possible.

NOTE: when specifying a path with a different Windows drive letter, always use the doubled backslashes. For example, `E:/mdtmp` and `file:/E:/mdtmp` will not work.

## 7. Remaining values that are reserved

```
com.softneta.pacs.gateway.pluginDir = plugins
com.softneta.meddream.patientStudyCountEnabled = true
logging.level.* = ERROR
logging.path = logging
logging.file = ${logging.path}/md.log

#com.softneta.meddream.pacs.configurations[0].type = dicom
#com.softneta.meddream.pacs.configurations[0].id = 1
#com.softneta.meddream.pacs.configurations[0].localAET = {Local AET}
#com.softneta.meddream.pacs.configurations[0].remoteAET = {Remote AET}
#com.softneta.meddream.pacs.configurations[0].host = {Remote PACS Host}
#com.softneta.meddream.pacs.configurations[0].port = {Remote PACS Port}
```

**Do not change, uncomment or comment these** unless instructed so by Softneta's support team.

2. Start the Java application manually with a console command "java -jar", for example, `java -jar MedDream-7.0.jar`.

**NOTE: for a more comprehensive support of compressed DICOM transfer syntaxes, the native libraries bundled with MedDream must be added when starting the application on Windows, or on Linux:**

1. open `libs.zip` and choose the folder suitable for your operating system. For example, `win-i686`;
2. extract file(s) from that folder to a new folder of your choice. For example, `C:\MedDreamPACS-Premium\services\lib`;
3. specify the path to this folder in the command that starts Java: `-Djava.library.path=PATH_TO_FOLDER`. Also the option `-jar` changes to `-cp` and another option `org.springframework.boot.loader.PropertiesLauncher` is added. For example, `java -cp MedDream-7.0.jar -Djava.library.path=C:\MedDreamPACS-Premium\services\lib org.springframework.boot.loader.PropertiesLauncher`.

You should first make sure everything works when the application is started manually. Then follow the next chapter for a more standalone setup.

**IMPORTANT:** If the Query Retrieve functionality is used:

**PHP:** The `tempDir` must use the default value.

**QR plugin:** `tempDir` and `cacheDir` must be the same.

## 7.2. Running as a service

### 7.2.1. Windows

1. In the directory of the Java application, copy `MedDream.NET2.exe` or `MedDream.NET4.exe` (depends on installed .NET version) to `MedDream.exe`;
2. Open the `Command Prompt`;
3. Run the command `MedDream.exe install`;
4. Start the service from the Windows service manager.

NOTE: `MedDream.NET2.exe` is used in case the .NET Framework 2.0 is installed meanwhile the `MedDream.NET4.exe` is used for .NET Framework 4.0.

NOTE: if you need a native library from “libs.zip” mentioned earlier, then update `<arguments>...</arguments>` in `MedDream.xml` accordingly.

### 7.2.2. Linux (System-V init)

If your architecture is not 64-bit but 32-bit, then rename `{INSTALL_DIRECTORY}/meddream-jar-wrapper.i386` to `meddream-jar-wrapper`.

If your system is RedHat-like (Fedora, RHEL, etc), then copy `{INSTALL_DIRECTORY}/meddream.redhat` to `{INSTALL_DIRECTORY}/meddream`. If it's Debian-like (Ubuntu, Debian itself, etc), then copy `{INSTALL_DIRECTORY}/meddream.debian` to `{INSTALL_DIRECTORY}/meddream`.

Update this line in `{INSTALL_DIRECTORY}/meddream` with real installation directory:

```
BASEDIR={INSTALL_DIRECTORY}
```

Execute the following commands:

```
sudo ln -s {INSTALL_DIRECTORY}/meddream /etc/init.d/meddream
sudo chmod +x {INSTALL_DIRECTORY}/meddream {INSTALL_DIRECTORY}/meddream-jar-wrapper
sudo service meddream start
sudo chkconfig meddream on
```

As Debian-like systems don't have `chkconfig`, the last command can be replaced with `sudo update-rc.d meddream defaults`.

By changing the line `USER=root` in `{INSTALL_DIRECTORY}/meddream`, you can force a less privileged user that can still access DICOM files etc. However this will likely require updating the line `PIDF=/var/run/$PROG.pid`, too, as `/var/run` is usually a privileged location (a `chown`'ed subdirectory will suffice in that case).

### 7.2.3. Linux (systemd)

Create the file `/etc/systemd/system/meddream.service` with the following text:

```
[Unit]
Description=MedDream Viewer
After=syslog.target

[Service]
WorkingDirectory=/opt/meddream/services/
User=www-data
ExecStart=/usr/bin/java -Xmx=1024m -cp /opt/meddream/services/MedDream-7.0.jar -Djava.
↪library.path=/opt/MedDream/services/lib org.springframework.boot.loader.
↪PropertiesLauncher
SuccessExitStatus=143

[Install]
WantedBy=multi-user.target
```

Execute the following command for service autostart:

```
sudo systemctl enable meddream.service
```

## 7.2.4. Linux (upstart)

Create a file /home/{user name}/.config/upstart/meddream.conf with the following text:

```
description "MedDream Viewer"
respawn
exec java -Xmx=1024m -cp {INSTALL_DIRECTORY}/MedDream-7.0.jar -Djava.library.path=
↳{INSTALL_DIRECTORY}/lib org.springframework.boot.loader.PropertiesLauncher
```

# 8. Integration with PACS

## 8.1. Integration modes

MedDream can access studies from PACS using:

Method	PACS
Direct access to PACS database	PacsOne
	DCM4CHEE
	Conquest
	ClearCanvas
	Ayca
Web Access to DICOM Persistent Objects (WADO)	Any PACS that supports the WADO-URI service
DICOM 3.0 native interface	Any PACS
File system access	Not applicable (a PACS is not necessary)

For best performance, it is recommended to use direct access to PACS database whenever available.

**WARNING:** In the Direct access to PACS database mode, MedDream must either:

- be installed on the same host as PACS, or
- have access to studies' files.

When MedDream is installed on a different host, the following problem might arise. Some databases contain paths to DICOM files stored on the PACS host (not on some common network storage, etc) and it is not always possible to create identical mount points or network drive letters on the MedDream host. In this case adding the parameter `$foreign_path_mapping` (config.php) will help. Example:

```
$foreign_path_mapping = "/volume1/DICOMNL\\|O:\\\n/volume2/DICOMNL\\|P:\\\";
```

Here `/volume1/DICOMNL\` is replaced with `O:\` under which the network resource shared by a Linux system is mounted on a Windows system; in the same fashion `/volume2/DICOMNL\` is replaced with `P:\`. For best results, please take a look at `php-YYYYMMDD.log` for original paths (see [Troubleshooting](#)).

## 8.2. Direct access to PACS database mode

### 8.2.1. PacsOne

#### 8.2.1.1. PacsOne notes

**WARNING:** PacsOne and Apache must run as the same user because newer versions of PacsOne create subdirectories with permissions too strict for different users. Please ensure that on Linux operating systems PacsOne and Apache services use the same user or belong to the same group.

WARNING: PacsOne must store the received DICOM files in the “DicomPart10” format. The other format, “Native”, is not fully supported by HTML5 MedDream (and you might have problems with other DICOM-related software). This setting is chosen during installation and later can be verified as follows:

- Windows: in the registry, REG\_SZ StorageFormat under HKEY\_LOCAL\_MACHINE\SOFTWARE\RainbowFish Software\PacsOne\\${AeTitle}
- Linux: the setting StorageFormat in file(s) \*.ini near PacsOne.exe

Under Linux, PacsOne’s custom libjpeg conflicts with standard libjpeg needed by the GD2 extension on which MedDream might still depend. You will need to disable the antispam image in order to log in to PacsOne web interface (to do that, place an empty file “no.antispam.code” near PacsOne.exe). You also won’t be able to view DICOM images in PacsOne; on the other hand, MedDream adds a “Show” button to the web interface and thus can be used instead of the built-in viewer.

PacsOne doesn’t have an index on series.modality. If searches with a particular modality are unacceptably slow compared to the default set of checkboxes, then you’ll need to use the following MySQL command:

```
ALTER TABLE series ADD INDEX (modality);
```

For the Send to DICOM Library Anonymizer, Java 7 and up is required.

Since 3.06.1103.3003, a DICOMDIR viewer in the subdirectory “DICOMDIR” is automatically included on export. However, an alternative way is to use PacsOne’s “Include external viewer” functionality. If you use the latter, and still keep the viewer in the subdirectory “DICOMDIR”, MedDream Export function might fail due to same files included twice. The solution is to rename the subdirectory, or move it somewhere outside the MedDream directory tree.

Until 6.0, the HIS integration by Patient ID was using a certain kind of fuzzy matching that includes coerced values of Patient ID. For example, /?patient=12345 will also list studies with Patient ID “12345[some\_original\_value]”. In 6.0 this is turned off by default. If you still need the legacy behavior, then change STRICTLY\_MATCH\_PATIENT\_FROM\_ACTION in meddream/pacs/PacsImpIPacsone/Search.php as follows:

```
const STRICTLY_MATCH_PATIENT_FROM_ACTION = false;
```

Since MySQL 5.7, the ONLY\_FULL\_GROUP\_BY mode is on by default. However MedDream’s Search function uses an incompatible query by default. We advise to disable ONLY\_FULL\_GROUP\_BY. If that’s not possible, then an updated and a **much slower** query can be switched on by changing NO\_MODALITY\_AGGREGATE in meddream/pacs/PacsImpIPacsone/Search.php as follows:

```
const NO_MODALITY_AGGREGATE = false;
```

During an upgrade it’s recommended to use the included config.sample-pacsone.php as a template and then update the resulting config.php according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

### 8.2.1.2. MedDream configuration for PacsOne

1. Perform steps described in paragraphs either *Deployment under Windows operating systems* or *Deployment under Linux operating systems*. It is recommended to use PACSONE\_INSTALL\_DIR/php as Apache’s DocumentRoot.
2. From PACSONE\_INSTALL\_DIR/php/meddream Copy and Replace applet.php to PACSONE\_INSTALL\_DIR/php.
3. Edit APACHE\_HTDOSCS\_DIR/meddream/config.php file. Use config.sample-pacsone.php as a template. The file contains basic instructions.
4. Navigate to `http://127.0.0.1/pacsone/meddream/home.php` (use PacsOne users to log in) or use the “Show” button in the PacsOne web system.
 

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.
5. For more supported image types, see *Deploying the Java-based core*.



## 8.2.2. DCM4CHEE

### 8.2.2.1. DCM4CHEE notes

WARNING: compatibility is tested only with DCM4CHEE 2.4.17, 4.3.0-Alpha4 and 5.6.0. Some (but not all) newer 5.x versions are tested occasionally.

WARNING: DCM4CHEE v4 and v5 (dcm4chee-arc) have been tested with MySQL and Oracle only.

WARNING: DCM4CHEE v2 has been tested with MySQL, MS SQL Server and PostgreSQL only.

WARNING: the integrated Reports functionality is so far implemented only for MySQL and Oracle. There are no schema files for MSSQL or PostgreSQL and existing queries are incompatible with these DBMSes.

For the Send to DICOM Library Anonymizer, Java 7 and up is required.

DCM4CHEE v4 requires that the path prefix (`filesystem.fs_uri`, default value specified in LDAP as `dcmInitFileSystemUri`) contains a URI scheme. MedDream supports the following syntax:

URI	Path	Applicable to operating system
<code>file:///dir/subdir1</code>	<code>/dir/subdir1</code>	all
<code>file:/dir/subdir2</code>	<code>/dir/subdir2</code>	all
<code>file:///&lt;drive&gt;:/subdir1</code>	<code>&lt;drive&gt;:/subdir1</code>	all
<code>file:/&lt;drive&gt;:/subdir2</code>	<code>&lt;drive&gt;:/subdir2</code>	all
<code>file://host/share/dir</code>	<code>//host/share/dir</code>	Windows
<code>file:///host/share/dir</code>	<code>/host/share/dir</code>	Windows
<code>file:///host/share/dir</code>	<code>//host/share/dir</code>	Windows

MedDream is able to automatically include a DICOMDIR viewer in each disc of exported studies. To use that, simply place the viewer files in a subdirectory named "DICOMDIR". You must also ensure that a file named "autorun.inf" exists in that directory, otherwise contents won't be included; an empty autorun.inf is suitable if the viewer somehow doesn't contain it.

During an upgrade it's recommended to use the included `config.sample-dcm4chee.php` or `config.sample-dcm4chee-arc.php` as a template and then update the resulting `config.php` according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

### 8.2.2.2. MedDream configuration for DCM4CHEE

1. Perform steps described in paragraphs either *Deployment under Windows operating systems* or *Deployment under Linux operating systems*.
2. Edit `APACHE_HTDOCS_DIR/meddream/config.php` file. Use `config.sample-dcm4chee.php` or `config.sample-dcm4chee-arc.php` as a template. The files contain basic instructions.
3. Add "studynotes" table to DCM4CHEE database from the schema file `reports-dcm4chee-mysql.sql` if you're using MySQL. For `dcm4chee-arc/dcm4chee-arc-lite` and Oracle, use `reports-dcm4chee-arc-oracle.sql` instead.
4. Add "attachments" table to DCM4CHEE database from the schema file `attachments-dcm4chee-mysql.sql` if you're using MySQL. For `dcm4chee-arc dcm4chee-arc-lite` and Oracle, use `attachments-dcm4chee-arc-oracle.sql` instead.
5. Navigate to `http://127.0.0.1/meddream/home.php` and use database accounts to log in. With DCM4CHEE 2.x, its internal accounts like "admin" or "user" are supported, too.

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.

6. For more supported image types, see *Deploying the Java-based core*.

## 8.2.3. ClearCanvas

### 8.2.3.1. ClearCanvas notes

WARNING: ClearCanvas support was last tested with MedDream 5.4. Pre-release tests with `$pacs = "ClearCanvas"` are rarely performed. Please contact [info@softneta.com](mailto:info@softneta.com) if you would like to use this integration mode.

WARNING: only the older ClearCanvas v2 is supported. We didn't test MedDream with ClearCanvas 3+.

Direct integration into IIS is not possible any more, as MedDream still does not support CGI/FastCGI, ISAPI support was removed in PHP 5.3 and we dropped support for 5.2. `php5.2_meddream.dll` from an older version will not work. Your only option is to host MedDream in Apache (running on a different port) and use IIS as a reverse proxy. Please contact [info@softneta.com](mailto:info@softneta.com) for instructions.

The traditional `php_mssql.dll` is officially deprecated and might not work with newer versions of PHP and SQL Server. Its replacement, the "Microsoft SQL Server Driver for PHP" (`$dbms = 'SQLSRV'` in `config.php`), can be downloaded from Microsoft free of charge. It contains many flavors of the driver; the ones supported by MedDream are non-PDO and TS. The v2.0 driver also requires the "Microsoft SQL Server 2008 R2 Native Client". Drivers older than v1.1 might be incompatible.

MedDream is able to automatically include a DICOMDIR viewer in each disc of exported studies. To use that, simply place the viewer files in a subdirectory named "DICOMDIR". You must also ensure that a file named "autorun.inf" exists in that directory, otherwise contents won't be included; an empty `autorun.inf` is suitable if the viewer somehow doesn't contain it.

During an upgrade it's recommended to use the included `config.sample-clearcanvas.php` as a template and then update the resulting `config.php` according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

### 8.2.3.2. MedDream configuration for ClearCanvas

1. Perform steps described in the paragraph *Deployment under Windows operating systems*.
2. Edit `APACHE_HTDOCS_DIR/meddream/config.php` file. Use `config.sample-clearcanvas.php` as a template. The file contains basic instructions.

You will likely need to specify a port for database connection, like `localhost,1433`.

3. Navigate to `http://server IP address:Apache port/meddream/home.php` (do not forget the actual port that Apache runs on, and use database accounts to log in).

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.

If the 'MSSQL' driver fails to log in, try installing the "Microsoft SQL Server 2008 R2 Native Client". If even that didn't help, change `$dbms` to 'SQLSRV' and manually install the "Microsoft SQL Server Driver for PHP". The file that must be added to PHP extensions and `php.ini`, depends solely on your PHP version as other two choices are "non-PDO" and "TS".

4. For more supported image types, see *Deploying the Java-based core*.

## 8.2.4. Conquest

### 8.2.4.1. Conquest notes

MedDream supports Conquest only with MySQL and SQLite v3 databases at the moment.

MedDream does not support the "V2 (allows NKI compression)" image format. All images that were received by Conquest with this setting on, including those of the example patient "HEAD EXP2", will be unusable.

For the Send to DICOM Library Anonymizer, Java 7 and up is required.

MedDream is able to automatically include a DICOMDIR viewer in each disc of exported studies. To use that, simply place the viewer files in a subdirectory named "DICOMDIR". You must also ensure that a file named "autorun.inf" exists in that directory, otherwise contents won't be included; an empty `autorun.inf` is suitable if the viewer somehow doesn't contain it.

During an upgrade it's recommended to use the included `config.sample-conquest.php` as a template and then update the resulting `config.php` according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

#### 8.2.4.2. MedDream configuration for Conquest

1. Perform steps described in paragraphs either *Deployment under Windows operating systems* or *Deployment under Linux operating systems*.
2. Edit `APACHE_HTDOCS_DIR/meddream/config.php` file. Use `config.sample-conquest.php` as a template. The file contains basic instructions.
3. Navigate to `http://127.0.0.1/meddream/home.php` (use database accounts to log in).

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.

4. For more supported image types, see *Deploying the Java-based core*.

### 8.2.5. Aycan

#### 8.2.5.1. Aycan notes

MedDream is able to automatically include a DICOMDIR viewer in each disc of exported studies. To use that, simply place the viewer files in a subdirectory named "DICOMDIR". You must also ensure that a file named "autorun.inf" exists in that directory, otherwise contents won't be included; an empty `autorun.inf` is suitable if the viewer somehow doesn't contain it.

During an upgrade it's recommended to use the included `config.sample-aycan.php` as a template and then update the resulting `config.php` according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

Installations of *aycan store* use Nginx as a web server. MedDream, however, does not support the FastCGI mechanism by which PHP integrates into Nginx. The following is needed (please contact [info@softneta.com](mailto:info@softneta.com) for detailed instructions):

- install Apache,
- run it on different ports (for example, 82 and 8443),
- add its user to the usergroup of the PACS process for access to DICOM files (for example, `adduser www-data users`),
- (optional) set up a reverse proxy in Nginx for more convenient URLs.

#### 8.2.5.2. MedDream configuration for Aycan

1. Perform steps described in the paragraph *Deployment under Linux operating systems*.
2. Edit `APACHE_HTDOCS_DIR/meddream/config.php` file. Use `config.sample-aycan.php` as a template. The file contains basic instructions.
3. Navigate to `http://server IP address:Apache port/meddream/home.php` (do not forget the actual port that Apache runs on, and use database accounts to log in).

From a security standpoint MedDream should use dedicated less-privileged database accounts. Minimum privileges are: SELECT on tables "stustore" and "serstore", SELECT and UPDATE on the table "session".

4. For more supported image types, see *Deploying the Java-based core*.

## 8.3. Web Access to DICOM Persistent Objects (WADO) mode

MedDream uses the `dcmqr` utility from `dcm4che2 Toolkit` to make all queries to the target PACS. This way any PACS that talks DICOM protocol is supported. DICOM images, however, are downloaded through WADO-URI interface so the target PACS must support it. Additionally, `allow_url_fopen` (`php.ini`) must remain enabled.

For the Send to DICOM Library Anonymizer, at least Java 7 is required.

MedDream is able to automatically include a DICOMDIR viewer in each disc of exported studies. To use that, simply place the viewer files in a subdirectory named “DICOMDIR”. You must also ensure that a file named “autorun.inf” exists in that directory, otherwise contents won’t be included; an empty autorun.inf is suitable if the viewer somehow doesn’t contain it.

During an upgrade it’s recommended to use the included config.sample-wado.php as a template and then update the resulting config.php according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

To configure MedDream query images using WADO mode, following steps must be performed:

1. Perform steps described in paragraphs either *Deployment under Windows operating systems* or *Deployment under Linux operating systems*.
2. Edit `APACHE_HTDOSCS_DIR/meddream/config.php` file. Use `config.sample-wado.php` as a template. The file contains basic instructions.
3. Restart Apache
4. Navigate to `http://127.0.0.1/meddream/home.php` and simply press the “Login” button. There is no password. Any non-empty user name is suitable. The user name “root” additionally enables the settings button.

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.

5. Usually the target PACS must be additionally configured to accept queries from the AET named “MEDDREAM”. You can assign any port, even the default 104, as `dcmqr` normally does not bind to a particular port and most PACSes do not require one when accepting a connection.
6. For more supported image types, see *Deploying the Java-based core*.

## 8.4. DICOM mode

MedDream uses the `dcmqr` utility from `dcm4che2` Toolkit to make all queries to the target PACS. DICOM images are downloaded in a similar fashion through the `dcmrcv` utility. This way any PACS that talks DICOM protocol is supported.

`dcmqr` identifies itself as “MEDDREAM” and is not required to bind to a particular port (though it’s able to do that). `dcmrcv` **must** bind to a port (we recommend 11116) and for simplicity it uses the same AET, “MEDDREAM”. In general,

- the target PACS must be configured to accept this AET on port 11116,
- `$db_host` should contain **only** the AET (without IP and port),
- `$dcm4che_recv_aet` must contain a full connection string with AET and port.

The `dcmrcv` utility will automatically start (if needed) in the background every time when someone logs in. Afterwards it is managed by a particular Apache child process that started it. When this process finishes, `dcmrcv` will stop as well, therefore concurrent sessions might suddenly encounter the “C-MOVE operation failed” error. This normally occurs only when stopping the entire web server. You should leave `MaxConnectionsPerChild` (`httpd.conf`) with a default value (zero), or set it to a large enough value, to maximize the lifetime of `dcmrcv`.

Since 5.x, the `RetrieveEntireStudy` mode is on by default. When opening a study, if at least one image is missing from the cache, the entire study is downloaded in the background – this is a lot faster. In case the old behavior (download a single image at once) is preferred, you can switch it back on:

1. In `php.ini`, change or add: `meddream.retrieve_entire_study = 0` (the default value is 1)
2. Restart Apache.

Only a limited support for Big Endian files is present. It causes some PACSes to consider the `A-ASSOCIATE-AC` response from `dcmrcv` invalid, and no files can be received at all. In that case your only option is to turn Big Endian support off, by adding the following to `php.ini`:

```
meddream.dcmrcv_cmdline="?1 -dest cached?2"
```

To configure MedDream using DICOM mode, following steps must be performed:

1. Perform steps described in paragraphs either *Deployment under Windows operating systems* or *Deployment under Linux operating systems*.
2. Edit `APACHE_HTDOCS_DIR/meddream/config.php` file. Use `config.sample-dicom.php` as a template. The file contains basic instructions.

`$db_host` and `$dcm4che_rcv_aet` may contain the same AET. The `@IP:PORT` part must be present in `$dcm4che_rcv_aet` only. It is better to avoid it in `$db_host`, however if you need it, then give a different value.

3. Navigate to `http://127.0.0.1/meddream/home.php` and simply press the “Login” button. There is no password. Any non-empty user name is suitable. The user name “root” additionally enables the settings button.

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.

4. Usually the target PACS must be additionally configured to accept queries from, and send images to, the AET named “MEDDREAM” on port 11116. If that port is already used on your system, choose a different one and specify it in `$dcm4che_rcv_aet`.
5. For more supported image types, see *Deploying the Java-based core*.
6. Some PACSes require that a Study Date search key is sent on every C-FIND request. You can configure a sufficiently wide range of dates in `meddream/settings/system.json`, parameters are called `studyMinDate` and `studyMaxDate` (date separators are not needed). For example,

```
{
  "studyMinDate": "19000101",
  "studyMaxDate": "21000101",
  "usePacsGwDirectly": false,
  "features": {
    "patientHistory": true,
    "keyObjects": true
  }
}
```

## 8.5. File system access mode

This MedDream mode allows to view single DICOM files, or directories containing files of a single study. Limitations are as follows:

- The path to a single file, or to a directory, will be relative. Absolute paths are not allowed due to security concerns. It is also not recommended to specify a drive root as a base directory. The simplest form of HIS integration results in paths visible for everyone, hence integration via POST requests is advised.
- If a directory is specified, then DICOM files are collected directly below it (subdirectories are not supported). Contents of the directory are sorted by name in ascending order and then parsed. All files must have the same Study UID; the value in the first file is treated as a reference and remaining files with different UIDs are ignored. Consequently, if the directory contains multiple studies, the outcome depends solely on file names and their collation according to the system-default locale.
- There is no search functionality. **A HIS (or equivalent application) must track studies/images together with corresponding paths, and offer hyperlinks to MedDream.** The interactive login, however, can still be used for the Settings dialog, provided that you log in as “root” (no password required). Alternatively, you can enable the dialog for all users, by changing `SHOW_USER` in `external.php` to “root”.
- The “Save Images...” function does not support the “Active Series” choice. However the user interface won’t warn about this, you’ll simply get `images.zip` which is a text file and contains an error message.

For the Send to DICOM Library Anonymizer, at least Java 7 is required.

During an upgrade it's recommended to use the included `config.sample-filesystem.php` as a template and then update the resulting `config.php` according to the old one. This will prevent misunderstandings like variable names which were suddenly changed.

To configure MedDream using File System mode, following steps must be performed:

1. Perform steps described in paragraphs either *Deployment under Windows operating systems* or *Deployment under Linux operating systems*.
2. Edit `APACHE_HTDOCS_DIR/meddream/config.php` file. Use `config.sample-filesystem.php` as a template. The file contains basic instructions.

You must specify `$archive_dir_prefix` – a base directory for all allowed paths.

3. Test the functionality by browsing to

```
http://127.0.0.1/meddream/?file=PATH_TO_YOUR_FILE
```

For example, the directory `C:\PACS\dcm4chee\server\default\archive` contains DCM4CHEE 2.x archive tree, with deeper levels named like `year\month\day\hour\...`. This directory is specified by `$archive_dir_prefix`. Then `PATH_TO_YOUR_FILE` could be `2012\2\4\0\43D7AA94\2569DF62\9242C40A` or, in JavaScript code, `2012\2\4\0\43D7AA94\2569DF62\9242C40A`.

If you get an HTTP 500 error, please check *Apache configuration* once more as likely some Apache module is missing.

4. For more supported image types, see *Deploying the Java-based core*.

## 8.6. Plugins for Java-based core

As of 7.0, support for some PACSes has been added to Java, in form of PACS plugins. It is disabled by default; to enable, make sure `meddream/settings/system.json` (created automatically during the first startup) contains `"usePacsGwDirectly":true`.

Multiple PACSes can be defined, then used in the Search window to open studies. Examples of configuration options are listed below.

NOTE: Some functions (namely, Export/Burn, Forward and Reporting, also some aspects of HIS integration) are still PHP-based at the moment. Consequently, their support depends solely on `$pacs` in `config.php` as before. Multiple PACSes will introduce additional issues: for example, the study opened from a different PACS might be missing on the PACS that is configured at PHP side.

1. PacsOne plugin

- Example: `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].type=PacsOnePlugin`  
Use this specific value of `type` when connecting to PacsOne.
- Example: `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].id=PacsOne`  
Identifies the plugin/PACS pair in the search window drop-down menu. You can connect to multiple PacsOne instances by using different values of `id` and remaining options.
- Example: `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].searchApiEnabled=true`  
Specifies if this plugin/PACS pair returns a study list for the Search window.
- Example: `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].imageApiEnabled=true`  
Specifies if this plugin/PACS pair retrieves image data.
- Example: `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].url=jdbc:mysql://127.0.0.1:3306/PACS_DB_NAME`

JDBC connection string. Only MySQL is supported.

**NOTE:** SSL requirements can be disabled adding the following arguments:

- **verifyServerCertificate=false** Disables Server certificate verification.
- **useSSL=false** Disables SSL usage.
- **requireSSL=false** Disables the SSL requirement.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].username=UserName`

Username for connecting to the database.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].password=UserPassword`

Password for connecting to the database.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].mappedStorageLocation=D:/DICOM=Y:\\DICOM1|E:/DICOM=Y:\\DICOM2`

**(optional)** Defines replacement paths for MedDream installed on a different host than the PACS and accessing images through mapped network drives.

## 2. DCM4CHEE 2.x plugin

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].type=Dcm4chee2Plugin`

Use this specific value of `type` when connecting to DCM4CHEE 2.x.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].id=Dcm4chee2`

Identifies the plugin/PACS pair in the search window drop-down menu. You can connect to multiple DCM4CHEE instances by using different values of `id` and remaining options.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].searchApiEnabled=true`

Specifies if this plugin/PACS pair returns a study list for the Search window.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].imageApiEnabled=true`

Specifies if this plugin/PACS pair retrieves image data.

- **Examples:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].url=jdbc:mysql://127.0.0.1:3306/PACS_DB_NAME, url=jdbc:postgresql://127.0.0.1:5432/PACS_DB_NAME`

JDBC connection string. Supported DBMSes are MySQL and PostgreSQL.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].username=UserName`

Username used to connect to the database.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].password=UserPassword`

Password used to connect to the database.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].charset=utf8`

**(optional)** Defines the character set used.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].enableStatistics=false`

**(optional)** `true` enables query statistics such as query time and so on. You might also want to set `logging.level.org.hibernate.stat=DEBUG` in `application.properties` to get more details.

### 3. DCM4CHEE 5 (dcm4chee-arc-lite) plugin

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].type=Dcm4chee5Plugin`

Use this specific value of `type` when connecting to DCM4CHEE 5.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].id=Dcm4chee5`

Identifies the plugin/PACS pair in the search window drop-down menu. You can connect to multiple DCM4CHEE instances by using different values of `id` and remaining options.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].searchApiEnabled=true`

Specifies if this plugin/PACS pair returns a study list for the Search window.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].imageApiEnabled=true`

Specifies if this plugin/PACS pair retrieves image data.

- **Examples:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].url=jdbc:mysql://127.0.0.1:3306/PACS_DB_NAME, url=jdbc:postgresql://127.0.0.1:5432/PACS_DB_NAME`

JDBC connection string. Supported DBMSes are MySQL and PostgreSQL.

In case of connection problems it might be necessary to add some more parameters to this string: `useJDBCCompliantTimezoneShift=true` for a time zone shift, `serverTimezone=UTC` for a specific timezone, `useSSL=false` to disable SSL, etc. (The first parameter-value pair is preceded by “?”, the remaining pairs – by “&”.)

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].username=UserName`

Username used to connect to the database.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].password=UserPassword`

Password used to connect to the database.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].dialect=MySQL`

Set database dialect. Supported values are `MySQL` and `PostgreSQL`. (Somewhat duplicates the `url` parameter, this is by design.)

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].modalityAggregate=true`

Attempt to collect all modalities for every study in the search results.

`false` might improve performance in case of MySQL with corresponding configuration, however then only a single modality from some series will be visible.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].storage=fs1=c:\\mnt\\NAS1;fs2=c:\\mnt\\NAS2`

Filesystem root directories, as configured in the LDAP tree via entries named “`dcmStorageID`”. The plugin is unable to read this configuration from LDAP at the moment.

### 4. Orthanc plugin



- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].type=orthancPlugin`

Use this specific value of `type` when connecting to Orthanc.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].id=Orthanc`

Identifies the plugin/PACS pair in the search window drop-down menu. You can connect to multiple Orthanc instances by using different values of `id` and remaining options.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].searchApiEnabled=true`

Specifies if this plugin/PACS pair returns a study list for the Search window.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].imageApiEnabled=true`

Specifies if this plugin/PACS pair retrieves image data.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].baseUrl=http://127.0.0.1:8042`

Base URL that points to the root of Orthanc Web API (`/tools/find`, `/instances` etc will be appended automatically).

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].username=UserName`

Username for Basic Authentication.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].password=UserPassword`

Password for Basic Authentication.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].dicomCacheDirectory=C:\\DICOM`

**(optional)** A directory for caching images. Without this setting a file is downloaded anew every time and it's not possible to examine its contents. Multiple instances of this plugin should have different directories.

**NOTE:** only directories below `${com.softneta.meddream.tempDir}/cache` are cleaned automatically. A different path will require some third-party solution for free space management.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].strictSearchIsEnabled=true`

**(optional)** Use exact matching by Patient ID in the Search window.

## 5. Query/Retrieve plugin

This is the equivalent of `$pacs = "DICOM"` in `config.php`.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].type=QRPlugin`

Use this specific value of `type` when connecting to a PACS over the DICOM 3.0 protocol.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].id=QRPlugin`

Identifies the plugin/PACS pair in the search window drop-down menu. Multiple instances of this plugin, however, are not recommended at the moment.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].searchApiEnabled=true`

Specifies if this plugin/PACS pair returns a study list for the Search window.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].imageApiEnabled=true`

Specifies if this plugin/PACS pair retrieves image data.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].remoteAET=PACS`

The AE Title of the remote machine.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].remoteHost=127.0.0.1`

IP address or hostname of the remote machine.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].remotePort=104`

Port number of the remote machine.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].localAET=MEDDREAM`

The AE Title of the local DICOM C-STORE receiver.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].localHost=127.0.0.1`

The IP address at which the local receiver listens. Currently it's not possible to leave an empty string and bind to all available addresses.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].localPort=11116`

Port number of the local receiver.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].dcmRcvPath=${com.softneta.meddream.configRoot}/dcm4che/bin/dcmrcv.bat`

Path to `dcmrcv.bat` (or `dcmrcv` under Linux). Both are available in the directory `meddream/dcm4che/bin`.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].cacheDir=C:\\DICOM`

A directory for receiving images.

**NOTE:** only directories below `${com.softneta.meddream.tempDir}/cache` are cleaned automatically. A different path will require some third-party solution for free space management.

## 6. DCMSYS plugin

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].type=dcmsys`

Use this specific value of `type` when connecting to DCMSYS Router.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].id=default`

Name of storage as defined in DCMSYS Administrator Site.

Multiple occurrences of `type=dcmsys` with different `id` is a standard way for supporting more than one storage.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].searchApiEnabled=true`

Specifies if this plugin/PACS pair returns a study list for the Search window.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].imageApiEnabled=true`

Specifies if this plugin/PACS pair retrieves image data.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].qidoUrl=https://127.0.0.1:8282/api/router/qido-rs`

Base URL for QIDO-RS requests used when searching for studies.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].wadoUrl=https://127.0.0.1:8282/api/router/wado-rs`

Base URL for WADO-RS requests used to access study metadata.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].dicomWadoUrl=https://127.0.0.1:8282/WADO`

Base URL for WADO-URI requests used to retrieve DICOM images.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].loginUrl=https://127.0.0.1:8282/api/login`

URL for the login endpoint.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].username=UserName`

Username for the login endpoint.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].password=UserPassword`

Password for the login endpoint.

- **Example:** `com.softneta.meddream.pacs.configurations[PLUGIN_INDEX_NUMBER].dicomCacheDirectory=/MedDreamTemps/temp`

**(optional)** A directory for caching images. Without this setting a file is downloaded anew every time and it's not possible to examine its contents. Multiple instances of this plugin should have different directories.

NOTE: only directories below `${com.softneta.meddream.tempDir}/cache` are cleaned automatically. A different path will require some third-party solution for free space management.

## 9. Image Access from hospital information system (HIS)

### 9.1. Generic HIS integration

HTTP POST or GET methods can be used to open a study in MedDream. The table below describes attribute types for study query.

Attribute	URI	Availability
Study UID	URL?study=STUDY_UID	Unsupported by File System
Accession Number	URL?accnum=ACC_NO	Unsupported by File System. Supported in external.sample-dcm4chee.php, external.sample-conquest.php, external.sample-pacsone.php, external.sample-rssdimed-pacsone.php
Patient ID	URL?patient=PATIENT_UID	Unsupported by File System
Patient ID and Accession Number	URL?patient=PATIENT_UID &accnum=ACCESSION_NO	Supported only for PacsOne in external.sample-azimuth-pacsone.php
Series UID	URL?series=SERIES_UID	PacsOne and DCM4CHEE only, external.sample-rssdimed-*.php required
Image UID	URL?image=IMAGE_UID	Supported in external.sample-rssdimed-pacsone.php, external.sample-dcm4chee.php, external.sample-rssdimed-dcm4chee.php
File Name	URL?file=PATH_TO_FILE	File System only

The most popular example: <http://localhost/meddream/index.php?study=1.2.392.200036.9107.500.110113>

To enable study query from HIS, the following steps must be performed:

1. Among files external.sample-\*.php, find the one dedicated to your PACS and rename/copy it to external.php.

external.sample-FULL.php is not recommended as it gets only compatibility fixes (no feature or security updates). It might be completely removed in the future. Please use a more specific example and email [info@softneta.com](mailto:info@softneta.com) in case the "FULL" example contains something important to your installation while the more specific one doesn't.

2. Modify first lines of external.php that look similar to these:

```
define("SHOW_ENABLED", true); // true: HIS integration is enabled; ↵
↵false: disabled
define("SHOW_DB", "dbname"); // database name
define("SHOW_USER", "user"); // a database user
define("SHOW_PASSWORD", "password"); // corresponding password
```

3. To open the Flash Viewer instead of HTML Viewer, add &htmlMode=off to the URL:

<http://localhost/meddream/index.php?study=1.2.392.200036.9107.500.110113&htmlMode=off>

**WARNING:** do *not* use an "alternative" form <http://localhost/meddream/swf/index.php?study=...>, as many, or even all, features described in this document will not work.

4. To add the study into the viewer opened in an existing browser tab, use addStudy.php instead of index.php:

<http://localhost/meddream/addStudy.php?study=1.2.392.200036.9107.500.110113&viewer=swf&token=363b122c528f54df4a0446b6bab05515>

<http://localhost/meddream/addStudy.php?study=1.2.392.200036.9107.500.110113&viewer=html&token=363b122c528f54df4a0446b6bab05515>

NOTE: the parameter "token" is optional by default. It becomes mandatory only with certain versions of external.php (external.sample-dcm4chee.php, external.sample-rssdimed-pacsone.php) and only if they configure a non-empty SHOW\_SSO\_ADDR.

When used this way, the viewer will contain multiple studies as if they were opened from Search without closing the browser tab containing the viewer.

Copy the following to an HTML file in the MedDream root directory, then update the URL near end of the file.

```

<!DOCTYPE html>
<html lang="lt">
  <head>
    <meta charset="utf-8">
    <title>Eli-MedDream</title>
  </head>
  <body>
    <button target="_blank" onclick="showImageUUID('1.2.840.113619.2.98.2188.
↪1490859935.0.2857','flash','token')">Image 1</button>
    <button target="_blank" onclick="showImageUUID('1.2.840.113619.2.98.2188.
↪1490859935.0.2857','html','token')">Image 2</button>
    <script>
      function showImageUUID(uuid,n,token) {
        window.open("http://127.0.0.1/addStudy.php?study=" + uuid + "&viewer="
↪ n + "&token=" + token, "MedDream", "");
      }
    </script>
  </body>
</html>

```

## 9.2. Specification of thumbnails service

Provides access to thumbnails of study images via HTTP GET/POST request. One request lists images in the study and the second request fetches a single JPEG image.

**IMPORTANT:** Will reuse existing login information if MedDream is already open in the same browser (other tab or window). For standalone use, you need to configure external.php.

Definition	Description
URL	MedDream root hyperlink without the query part (e.g., <a href="http://localhost/meddream/">http://localhost/meddream/</a> ). It is OK to include index.php or home.php.
image_uid	SOP Instance UID
study_uid	Study Instance UID
image_size	Limit of image width and height. The image will be resized so that the larger side won't exceed this value. Default value is 50, allowed range is 50 ... 4320.
data_result	Choice of images to list: <ul style="list-style-type: none"> <li>0 - first image from the first series</li> <li>1 - first image from every series</li> <li>2 - all images from the study</li> </ul> Default value is 0.

### Request: list available thumbnails

URL/getImageList.php?study=study\_uid&size=image\_size&result=data\_result

Returns a JSON-formatted array:

```

{
  "studyUID": study_uid,
  "thumbnails": [
    {
      "imageUID": image_uid_1,
      "url": URL/getThumbnail.php?image=image_uid_1&size=image_size
    },
    ...,
    {
      "imageUID": image_uid_N,
      "url": URL/getThumbnail.php?image=image_uid_N&size=image_size
    }
  ]
}

```

(continues on next page)

(continued from previous page)

```
]
}
```

In case of not found study, authentication failure or any other error, will return the HTTP 404 error with an empty Response Body.

### Request: get a thumbnail image

URL/getThumbnail.php?image=image\_uid&size=image\_size

Returns the raw JPEG stream with an appropriate Content Type header.

In case of not found image, authentication failure or any other error, will return the HTTP 404 error with an empty Response Body.

## 9.3. Specification of Live Stream

Addresses of live streams can be defined in the file `liveStreamList.json` in the MedDream installation directory (for example, near the file `Routes.php`). The configured items will appear in MedDream search results as studies.

**IMPORTANT:** a live stream address must be valid and accessible via VLC player on every client machine. You must make sure that the address is accessible to the client: port is open, antivirus or firewall software isn't blocking it, etc.

The file `liveStreamList.json` must contain valid JSON data.

Definition	Description
liveStream	Stream file block. Required.
id	Stream id. Short word. Required.
name	Stream short name.
modality	Set to "LIVE".
description	Detailed description of the stream, a single line.
sourceAE	Full source address that can be played in the VLC player.

### Example

```
{
  "liveStream": [
    {
      "id": "1",
      "name": "Room #2",
      "modality": "LIVE",
      "description": "Operating room in 2nd floor\ndasdasd",
      "sourceAE": "rtsp://192.168.1.1:554/session0.mpg"
    },
    {
      "id": "2",
      "name": "Room #3",
      "modality": "LIVE",
      "description": "Operating room in 3rd floor",
      "sourceAE": "rtsp://192.168.1.2:554/session0.mpg"
    }
  ]
}
```

## 9.4. Further reading

More details can be found in a separate document, `quick_install-HIS_integration.txt`.

## 10. Additional software

MedDream can be used in tandem with several external pieces of software that are listed below.

NOTICE: to get the required software, please contact [support@softneta.com](mailto:support@softneta.com).

### 10.1. 3D service

#### 10.1.1. 3D server configuration

The XStream HDVR WebSDK runs with default system configuration (non-SSL through port **8088**) with the most common quality and performance settings. Default behavior is described below and is changed by modifying the file `foviaserver/config/configuration.xml`. The server must be restarted for any changes to take effect. The system configuration and default settings are described below:

```
<nodePort>8088</nodePort>
<useSSL>0</useSSL>
<jpegInteractive>50</jpegInteractive>
<jpegFinal>95</jpegFinal>
```

#### 10.1.2. Network settings (ports and SSL)

`nodePort` and `useSSL` above specify the network protocol and port to be used. Separate port configuration settings are used for SSL and non-SSL, although only one can be used at one time.

The port can be changed to any valid unused port, such as 9123. After restarting your server, validate that the test app “helloFovia” still works correctly after the corresponding reconfiguration (`<FoviaServerIPAddress>...:9123/version...`).

To enable SSL, set `useSSL` to 1 and restart your server. You can test the HTTPS protocol on the URL `https://localhost:8088/version` which uses the supplied test certificate. In a production environment, replace the supplied demo files `server.crt` and `server.key` in the directory `foviaserver/ssl` with your company’s official certificate.

#### 10.1.3. Quality / Performance

Default values for quality and performance are explained below. Care must be taken when changing these settings since they affect all users on the system. There are two types of quality / performance metrics in the system: those that relate to network, and those that relate to the actual rendering (and ray casting).

The network quality and performance trade off affects how much bandwidth is required for transferring images over the network, and specifically the amount of JPEG compression for interactive and final quality images. This is largely independent of rendering quality that is described below.

There are two levels of JPEG compression: `jpegInteractive` is set to 50/100 and `jpegFinal` is set to 95/100. Typically during mouse interactions, many frames of quality `jpegInteractive` are generated (such as during the `render()` call during a `mouseMove` operation). These are quick rendering operations with more aggressively compressed JPEG images. On `mouseUp`, a `renderFinal()` operation is triggered and results in a frame of quality `jpegFinal`. The rendering engine will take longer to generate higher quality images, and JPEGs with less aggressive compression will be used for these.

The server version and configuration settings are displayed at **`http://localhost:8088/version`**.

The 3D service creates a 3D representation of the study. In order to use the 3D service the MedDream license must have the “3D” module and the following files have to be edited.

##### 10.1.3.1. `system.json`

In this file, update the parameter `threeDimensionServiceIntegration`:

```
{
  "threeDimensionServiceIntegration": [{
    "serverIP": "127.0.0.1",
    "port": "8088",
    "protocol": "http:",
    "pathPrefix": null,
    "pathPrefixReplaceTo": null
  }]
}
```

- serverIP

Address of the server where the 3D service is installed. Default value: "127.0.0.1".

- port

The set port of the 3D service. Default value: "8088".

- protocol

The transfer protocol used by the server. Possible values: **http**, **https**.

- pathPrefix

Used if the 3D server is on a different machine than the PACS and the latter keeps DICOM files locally. Sets the base path of the DICOM file storage folders (like D:\DICOM). Default value: null.

- pathPrefixReplaceTo

A local equivalent of `pathPrefix`. At least this contains the path on the 3D server machine where DICOM file storage shared on the PACS machine is mounted (like Z:\DICOM) . Default value: null.

### 10.1.3.2. Apache configuration file (httpd.conf)

If outside connections to the 3D service are not allowed, then the Apache configuration needs the following lines:

```
<IfModule proxy_module>
  <IfModule proxy_http_module>
    RewriteEngine On
    RewriteCond %{REQUEST_URI} ^/socket.io [NC]
    RewriteCond %{QUERY_STRING} transport=websocket [NC]
    RewriteRule /(.*) ws://{3D_SERVICE_IP}:{3D_SERVICE_PORT}/$1 [P,L]
    ProxyPass /socket.io http://{3D_SERVICE_IP}:{3D_SERVICE_PORT}/socket.io
    ProxyPassReverse /socket.io http://{3D_SERVICE_IP}:{3D_SERVICE_PORT}/socket.io
  </IfModule>
</IfModule>
```

**NOTE:** When SSL is used, change `ws` to `wss` and `http` to `https`.

## 10.2. Browser plugin

A plugin for the Chrome browser that expands a newly opened tab across several monitors. The URL is scanned for a entered keyword and if a match is found, a new window is created and expanded across the selected monitors. The plugin can be added using the Chrome extension tab.

NOTICE: to get plugin, please contact to [support@softneta.com](mailto:support@softneta.com)

## 10.3. OpenAM verification service

OpenAM is an open source access management and entitlements server platform. This replaces the standard MedDream verification service. Installation is explained in the next chapter.

NOTICE: to get prepared OpenAM package, please contact to [support@softneta.com](mailto:support@softneta.com)



# 11. MedDream and OAuth

MedDream 6.0 and later versions add a possibility to use OAuth for authentication and user rights management. Implementation is based on OpenAM.

The file `external-oauth2.php` provides client-side integration. It is based on `external.sample-pacsone.php` and is therefore to be used with `$pacs = 'PacsOne'` (`config.php`). However if you don't need HIS integration and authentication is enough, then this is a working authentication solution for configurations like `$pacs = 'DICOM'`.

## 11.1. Installation

Run OpenAM server on embedded Tomcat from the installation folder (`OPENAM_HOME`):

```
java -jar openam-server.jar
```

It will start server on default 8090 port. To run on specific port for example 8088:

```
java -jar openam-server.jar -port 8088
```

### OpenAM server

<http://host:port/openam>

### Initial setup

Make sure OpenAM server is started. During configuration enter:

OpenAM server url - <http://host:port>

Redirection URI of OAuth2 provider - [http://<meddream\\_host>:<meddream\\_port>/index.php](http://<meddream_host>:<meddream_port>/index.php)

### Windows

Run script:

```
OPENAM_HOME\scripts\config.bat
```

### Linux

Set access rules:

```
chmod a+x OPENAM_HOME/scripts/*.sh chmod 400 OPENAM_HOME/tools/admin/pwd.txt
```

Run script:

```
OPENAM_HOME/scripts/config.sh
```

### Remove configuration

Stop OpenAM server

### Windows

Run script:

```
OPENAM_HOME\scripts\remove-config.bat
```

### Linux

Run script:

```
OPENAM_HOME/scripts/remove-config.sh
```

### PacsOne users import

“mysql” **must** be accessible from any location.

### Windows

Run users extraction script:

```
OPENAM_HOME\scripts\extract-users.bat <users export file> <db host> <db user> <db password> <db name>
```

Real example:

```
OPENAM_HOME\scripts\extract-users.bat OPENAM_HOME\scripts\users 192.168.11.10 root password dev
```

Run users import script:

```
OPENAM_HOME\scripts\import-users.bat <users export file>
```

Real example:

```
OPENAM_HOME\scripts\import-users.bat OPENAM_HOME\scripts\users
```

## Linux

Run users extraction script:

```
OPENAM_HOME/scripts/extract-users.sh <users export file> <db host> <db user> <db password> <db name>
```

Real example:

```
OPENAM_HOME/scripts/extract-users.sh OPENAM_HOME/scripts/users 192.168.11.10 root password dev
```

Run users import script:

```
OPENAM_HOME/scripts/import-users.sh <users export file>
```

Real example:

```
OPENAM_HOME/scripts/import-users.sh OPENAM_HOME/scripts/users
```

## 12. MedDream and external proxies

After a successful login `index.php` (or `home.php`) redirects to itself in order to store credentials in the PHP session. By default the target URL depends on `$_SERVER['HTTPS']` and `$_SERVER['HTTP_HOST']`.

If MedDream is served by a machine inside a LAN, which is exposed to Internet by a proxy server only, then the browser will receive a URL containing the LAN-specific server name from `$_SERVER['HTTP_HOST']`, and likely won't be able to open it. In this case, set `DEFAULT_USE_PROXIED_HOST = true` in `meddream/WebTarget.php` and make sure the proxy server adds the `X-Forwarded-Host` HTTP header.

Some installations enforce HTTPS only for the “outside” connection: the browser uses HTTPS when connecting to the proxy, while the proxy uses HTTP when connecting to MedDream. Usually this is not a problem as the redirect to HTTP is converted to a HTTPS redirect. If the proxy doesn't handle this (HTTP isn't working at all), then you can set `DEFAULT_USE_PROXIED_PROTO = true` in `meddream/WebTarget.php` and add `X-Forwarded-Proto` header in the proxy server.

In a similar fashion `DEFAULT_USE_PROXIED_PORT = true` together with the `X-Forwarded-Port` header could be used to override the port. As `$_SERVER['HTTP_HOST']` normally includes the port, you might need to override both port and host.

## 13. Security considerations

**RATIONALE:** It is an old dilemma of using “security through obscurity”. As a matter of fact, most MedDream installations tend to have a few security holes due to non-paranoid Web administrators. The very publication of this knowledge makes every old/unmaintained MedDream installation an easy target. But, we must draw a line one day so that at least new installations are secure.

### 13.1. Search engines

If MedDream is exposed to the Internet, your installation might be found simply by searching for “MedDream” or “Softneta”. (Rebranding is possible but most customers don't need it.) Afterwards an attacker can check for typical misconfigurations as explained later in this document.

**Solution: set up the robots.txt file.**

This file is always hosted as `/robots.txt` (at the website root). If MedDream is also hosted there, the following robots.txt will be sufficient:

```
User-agent: *
Disallow: /

User-agent: AdsBot-Google
Disallow: /
```

A recommended read is <http://www.robotstxt.org/robotstxt.html>.

## 13.2. The subdirectory ‘log’

This subdirectory may contain files `php-YYYYMMDD.log` with sensitive patient data. Logs are not enabled by default, but might be needed during troubleshooting which sometimes takes days. Also one might forget to remove them (or disable logging) afterwards.

**Solution: disable Web access to this directory altogether.**

When using Apache, this is achieved by the following `.htaccess` file in this directory:

```
Order deny,allow
Deny from all
```

Apache 2.4+ requires a different syntax:

```
Require all denied
```

Reminder: If you need a restrictive `AllowOverride` directive on a related directory, its compatible values are “Limit Options” (first one allows `Order/Deny`, and the second one is for `Options -Indexes` in the next chapter). The equivalent for 2.4+ is “AuthConfig Options”. Furthermore, MedDream requires “Fileinfo” since 6.1 so you must add this one when changing `AllowOverride All` to a more detailed subset.

## 13.3. The subdirectory ‘temp’

The subdirectory “temp” is accessed via Web when viewing DICOM MPEG2 (Transfer Syntax UID = 1.2.840.10008.1.2.4.100) files, and when downloading an `.ISO/.burn` file after Export function finishes. It might contain temporary files and subdirectories with randomized names and DICOM/JPG/FLV data inside; every such file might reveal sensitive patient information. Some of these files are left undeleted when MedDream crashes, and some are just impossible to delete on time. Scripts `deleteTemp.*` can be regularly called by the operating system’s scheduler to minimize amount of rubbish, however the latter will not disappear completely.

**Solution: disable Web listing of this directory.**

When using Apache, this is achieved by the following `.htaccess` file in this directory:

```
Options -Indexes
```

Alternatively, the entire MedDream directory can be denied listing in the `<Directory>` configuration block.

Furthermore, if you do not intend to view MPEG2 or use the Export function (or MedDream doesn’t support them in your configuration), then simply disable Web access to this directory altogether, as with “log” subdirectory above.

## 13.4. printVersions.php

This file returns versions of major MedDream components **without requiring authentication**. It is not needed by MedDream itself and is rather dedicated to help investigations by service staff and automated deployments.

If you believe that versions might suggest known vulnerabilities and therefore speed up their search, then at least rename this file to something obscure.

## 13.5. DCM4CHEE 2.x

Official installation instructions offer some defaults that sometimes are left unchanged.

The default database name for 2.x is “pacbdb”. You also might be tempted to call it “dcm4chee”. For a knowledgeable attacker, both provide a clear suggestion which PACS is there and what to try next.

After a mindless installation there also will be some default user accounts:

- a database user “pacs” with password “pacs”. Perfectly suitable for MedDream;
- an internal user “admin” with password “admin” – suitable for MedDream and DCM4CHEE’s web interface;
- an internal user “user” with password “user” – suitable for MedDream and DCM4CHEE’s web interface.

Any of these accounts can be used to access sensitive patient data. We suggest to change passwords for all three, as soon as possible after the installation. The first one is used by DCM4CHEE to connect to the database, therefore you will also need to update the file server/default/deploy/pacs-\*-ds.xml (name depends on database used).

## 13.6. The subdirectory ‘scripts’

This subdirectory contain files with background jobs and related logs.

**Solution: disable Web access to this directory altogether.**

When using Apache, this is achieved by the following .htaccess file in this directory:

```
Order deny,allow
Deny from all
```

(Apache 2.2) or

```
Require all denied
```

(Apache 2.4+).

## 13.7. prepared.php

This file (since 6.1) allows to manage failed preparation jobs, examine DICOM tags and even download entire objects. Access is protected by a password (no user name) that is stored unencrypted in a configuration file. The default password becomes known to everyone who downloads the MedDream installation archive.

**Solution: change the password as soon as possible after the installation.**

File meddream/scripts/prep/config.php:

```
define('WEB_PASSWORD', 'the new password'); /* used by prepared.php, without_
↪user name */
```

## 13.8. Embedding into IFRAME etc

Availability to embed MedDream into FRAME, IFRAME or OBJECT containers allows untrusted websites to use MedDream in fraudulent manner (“clickjacking”). Since v6.2.1 this is restricted by default: only the server that hosts MedDream is allowed to wrap it with an additional container.

This is implemented by the X-Frame-Options header in AuthDB.php:

```

public function addXFrameOptionsHdr()
{
    /* alternatively you can configure the header globally in the webserver and
↳comment this
        line out
    */
    header('X-Frame-Options: sameorigin');
}

```

You can comment out the entire line to remove the restriction, or change `sameorigin` to `deny` in order to disallow any embedding.

**It is recommended to add the header to webserver's configuration instead.** In that case comment out the line above and add the following to the website configuration:

```
Header always set X-Frame-Options "sameorigin"
```

Other possible values of the header and their support in browsers can be found [here](#).

What if multiple hosts should be allowed to embed MedDream, while keeping others restricted? The mentioned place of `AuthDB.php`, or the webserver's configuration, can be used to add the [Content-Security-Policy header](#). Internet Explorer will need another header, `X-Content-Security-Policy: sandbox`.

## 14. Localization

MedDream is certified with three translations: English, Lithuanian, Russian. All three are bundled with MedDream, however only English is enabled by default. You can specify any available language or their combination in `meddream/config.php`. For example,

```

--- a/config.php      2018-10-12 11:30:52.663417900 +0300
+++ b/config.php      2018-11-06 15:22:48.085514800 +0200
@@ -120,7 +120,7 @@

         Example: "en,lt,ru"
     */
-    $languages= "en";
+    $languages= "en,lt,ru";

    /* default login credentials (for demonstration purposes etc)

```

The first language listed here will be offered by default on user's computer.

**WARNING:** adding a language beyond those three will invalidate MedDream certifications.

## 15. Rebranding

MedDream can be rebranded and use custom branding information (e.g., logo, product name, contact details).

Important things to consider:

- By default the software license does not allow rebranding of the About dialog. Please contact [info@softneta.com](mailto:info@softneta.com) for an updated license that is fully compatible with rebranding.
- The license can also define software as a module and, as a result, hide the following parts of the About dialog: "Contacts:" (Flash Viewer only), "Manufactured by:", "Medical device class:", "ID of the notified body:", "FDA Cleared:", the FDA and CE labels.
- Rebranded software does not check for recent versions; the version displayed in About dialog is always considered up to date.

- If the login form tells that `rebranding_configuration.json` is not valid, you should check its format with some JSON editor or online resource (for example, <http://jsonlint.com/>) and also confirm that the “site” attribute begins with “http”.

Configuration and related files must be located in the directory `APACHE_HTDOCS_DIR/meddream/rebranding/`.

The main configuration file is ‘`rebranding_configuration.json`’. It must contain a valid JSON object without comments. The file must be encoded in UTF-8.

JSON object attributes:

- `isRebranded`

Possible values: true, false

Default value: false

This attribute is **mandatory**. It allows to turn on/off rebranding without changing a lot of other attributes. If the configuration file is missing, or attributes are empty, then corresponding images and contact information are not displayed.

Example:

```
"isRebranded":true
```

- `productName`

Possible values: any text

Replaces the text “MedDream” in the bottom left corner of the login form, the “Product:” part of the About dialog, the title of the browser tab.

Example:

```
"productName": "MedDream"
```

- `productVersion`

Possible values: any text

Replaces the version in the bottom left corner of the login form, and the “Version:” part of the About dialog.

Example:

```
"productVersion": "1.1.1"
```

- `copyright`

Possible values: any text

Replaces the text “Copyright 2018 © Softneta” in the bottom left corner of the login form, and below search results.

Example:

```
"copyright": "Copyright 2018 © Softneta"
```

- `companyLogoFile`

Possible values: name (with extension) of an image file in the `./rebranding` subdirectory

Image format: PNG, JPG

Replaces the company logo in the About dialog.

Size of the default image is 134 x 28. The maximum width is 600 pixels, anything past this is *clipped*. Image height is not limited, however it will cause some repositioning of the dialog contents; a too high image might move the buttons outside the screen so the dialog becomes unusable. Make sure to check your customization on various client systems (different builds of Flash Player, different display DPI, etc) from where MedDream will be opened.

**Example:**

```
"companyLogoFile" : "companylogo.jpg"
```

(refers to file *APACHE\_HTDOCS\_DIR*/meddream/rebranding/companylogo.jpg)

- `productLogoFile`

Possible values: name (with extension) of an image file in the `./rebranding` subdirectory

Image format: PNG, JPG

Replaces the product logo in viewer's top left corner.

Size of default image is 167 x 37. The height can be up to 51 px, anything more is clipped. If the width exceeds 224 px, the thumbnail stripe will move to right. When your picture is near these limits, it might need checking on various client systems to ensure consistent behavior.

**Example:**

```
"productLogoFile" : "productlogo.jpg"
```

(refers to file *APACHE\_HTDOCS\_DIR*/meddream/rebranding/productlogo.jpg)

- `loginLogoFile`

Possible values: name (with extension) of an image file in the `./rebranding` subdirectory

Image format: PNG, JPG

Replaces the picture above the login form.

The width should be 290 px exactly (see `“.content”` in `assets/css/style.css`) as the image is left-aligned in a container having this width.

The height is not limited. Of course the entire login form might not fit on smaller screens, especially in mobile devices, due to excessive height.

**Example:**

```
"loginLogoFile" : "loginlogo.jpg"
```

(refers to file *APACHE\_HTDOCS\_DIR*/meddream/rebranding/loginlogo.jpg)

- `contacts` (Flash Viewer only)

Possible values: any text, line separators indicated by `\n` (end of line)

Replaces the “Contacts:” part of the About dialog.

The dialog has space for 5-6 lines of text and wrapping is enabled. Subsequent lines, either due to embedded breaks or wrapping, will display vertical scrollbar. As before, a large amount of text might need checking on various client systems.

An email address is made clickable by surrounding it by `email{ and }`. It will open in the default mail client.

A web address is made clickable, and of different text color, by surrounding it by `link{ and }`. It will open in the default Web browser. The default protocol scheme is `“http://”`, so add `“https://”` if you need HTTPS.

**Example:**

```
"contacts" : "company\naddress\ncountry\nemail{test@test.com}\nlink{www.web.com}"
```

1 line displays: company

2 line displays: address

3 line displays: country

4 line displays: test@test.com

(if clicked, opens the default email client and offers to compose an email to this recipient)

5 line displays: `www.web.com`

(if clicked, opens the default browser and navigates to this address)

- `companyName` (HTML Viewer only)
 

Possible values: any text, preferably single line.

Replaces the 1st line under "CONTACTS:" in the About dialog.
- `streetAddr` (HTML Viewer only)
 

Possible values: any text, preferably single line.

Replaces the 2nd line under "CONTACTS:" in the About dialog.
- `postCode, city` (HTML Viewer only)
 

Possible values: any text, preferably single line.

A combination of these (`postCode + , + city`) replace the 3rd line under "CONTACTS:" in the About dialog.
- `country` (HTML Viewer only)
 

Possible values: any text, preferably single line.

Replaces the 4th line under "CONTACTS:" in the About dialog.
- `email` (HTML Viewer only)
 

Possible values: any text, preferably single line.

The 5th line under "CONTACTS:" in the About dialog will display this text. A prefix "E-mail: " is always added to the displayed text. A prefix "mailto:" is always added to the underlying address.

If this line is clicked, MedDream opens the default email client and offers to compose an email to this recipient.
- `site` (HTML Viewer only)
 

Possible values: any text that begins with "http" (you can use "https", too)

The 6th line under "CONTACTS:" in the About dialog will display this text. The same text is always added as the underlying address.

If this line is clicked, MedDream opens a new browser tab and navigates to this address.
- `disclaimerText` (Flash Viewer only)
 

Possible values: any text, line separators indicated by `\n`

Adds a disclaimer pop-up message that is displayed every time the viewer is opened.

Example:

```
"disclaimerText": "my disclaimer text goes here"
```
- `defaultTheme` (HTML Viewer only)
 

Possible values: `red` (default), `blue`, `green` NOTICE: default theme can not be applied for the login page.

Example:

```
"defaultTheme": "blue"
```

A typical "all included" example of `rebranding_configuration.json`:



```
{
  "isRebranded":true,
  "productName":"MedDream",
  "productVersion":"1.1.1",
  "copyright":"Copyright 2017 © Softneta",
  "companyLogoFile":"companylogo.jpg",
  "productLogoFile":"productlogo.jpg",
  "loginLogoFile":"loginlogo.jpg",
  "contacts":"company\naddress\ncountry\nemail{test@test.com}\nlink{www.web.com}",
  "companyName":"company",
  "streetAddr":"address",
  "postCode":"postal code",
  "city":"city",
  "country":"country",
  "email":"test@test.com",
  "site":"http://www.web.com",
  "disclaimerText":"my disclaimer text goes here",
  "defaultTheme": "red"
}
```

Some **window names**, namely, “MedDream search”, “MedDream viewer”, “MedDream Settings” visible on the tab header, currently can’t be rebranded this way. The recommended workaround is to update the translation file. For example, the English translation meddream\locales\en\translation.6.1.1.json for MedDream 6.1.1 could be changed as follows:

```
--- translation.6.1.1.json.ORIG 2018-02-28 17:04:55.677247300 +0200
+++ translation.6.1.1.json      2018-02-28 17:12:45.988251300 +0200
@@ -21,7 +21,7 @@
     "DateTime": "Date Time",
     "ReceivedOn": "Received On",
     "SourceAE": "Source AE",
-    "Tittle": "MedDream search",
+    "Tittle": "MyProduct search",
     "About": "About",
     "LicenseAgreement": "License Agreement",
     "MoreThan": "Filtered more than 1000 records!",
@@ -72,7 +72,7 @@
     "Close": "Close",
     "Apply": "Apply",
     "Cancel": "Cancel",
-    "Tittle": "MedDream viewer",
+    "Tittle": "MyProduct viewer",
     "date": "Date:",
     "CloseStudy": "Close study",
     "CloseStudies": "Close studies",
@@ -239,7 +239,7 @@
     "notAvailable": "Share Dicom library not available when license do not allow_
↵version updates."
   },
   "settings": {
-    "Tittle": "MedDream Settings",
+    "Tittle": "MyProduct Settings",
     "On": "Enable",
     "Off": "Disable",
     "Rows": "Rows:",
```

---

## 16. Troubleshooting

### 16.1. Log files of the legacy backend

The directory `APACHE_HTDOCS_DIR/meddream/log` contains two kinds of log files:

- `php-YYYYMMDD.log` – log messages from PHP-based core
- `YYYYMMDD-HHISS-PPPPP.log` – log messages from the `php_meddream` module

By default this directory also contains the file `enabled_` that is ignored; it will come into effect after renaming to `enabled` (without the underscore).

The file `enabled` configures the log level, specified by a single digit 0 to 4 (more levels might be added in the future):

- 0 – off
- 1 – errors
- 2 – warnings
- 3 – information
- 4 – debug

It is recommended to leave the value “1” in new installations. Currently, however, messages from the lower levels might have not enough context information to be usable.

After changing the level, it is necessary to restart Apache so that `php_meddream` also reacts to the change. Otherwise only `php-YYYYMMDD.log` is affected.

### 16.2. Audit logs

Starting from 6.2, the directory `APACHE_HTDOCS_DIR/meddream/log` by default also contains files `audit-YYYYMMDD.log`. Their main purpose is to log access to specific backend endpoints for security auditing. But, in some cases their contents might provide additional context for messages from `php-YYYYMMDD.log`.

Please see `quick_install-HIPAA_audit_log.txt` for details.

### 16.3. Log files of the Java-based core

In the directory of the Java application, the files `MedDream.*.log` will be created. `MedDream.out.log` contains console output of the application.

To control the logging level, add the following line to `application.properties`:

```
logging.level.root=DEBUG
```

Alternatives to “DEBUG” are INFO, ERROR etc as per Log4j specification.

### 16.4. Browser download path selection

#### 16.4.1. Google Chrome

You can choose a location on your computer where downloads should be saved by default or pick a specific destination for each download.

- On your computer, open Chrome.
- At the top right, click More and then **Settings**.
- At the bottom, click **Advanced**.

- Under the “Downloads” section, adjust your download settings:
- To change the default download location, click **Change** and select where you’d like your files to be saved.
- If you’d rather choose a specific location for each download, check the box next to “Ask where to save each file before downloading”.

If you didn’t change your default download location, then Google Chrome downloads files to the following locations:

- **Windows 10, 8, 7 and Vista:** Users<username>Downloads
- **Mac:** /Users/<username>/Downloads
- **Linux:** /home/<username>/Downloads

### 16.4.2. Mozilla Firefox

**Note:** Changing the location of your downloads affects **all** downloaded files in this Web browser.

- Click the menu icon on the top right corner of the browser.
- Click **Preferences**.
- Click **General**.
- Click **Choose...** next to Save files to.

### 16.4.3. Microsoft Edge

- Open Microsoft Edge.
- Select **Settings and more** -> **Settings**.
- Under **Downloads**, select **Change**.
- In the dialog box, select a new location for your downloaded files.

### 16.4.4. Safari

To change the default download location of your Safari browser:

- Click on the “**Edit Menu**” -> **Preferences** -> **General tab**
- Locate the “**Save downloaded files to**” section, Click on “**Downloads**” > “**Other**”...
- Browse and indicate your new download location.

### 16.4.5. Internet Explorer 11

- click on the **gear icon** (top right corner of the window), and choose “**View Downloads**” from the menu
- click on the **Options** link in the bottom left corner
- click on the **Browse** button and pick another folder. Click **OK** once done.