

CTP Configuration for the Image Sharing Project

From MircWiki

This article describes how the CTP application can be configured for use in the RSNA Image Sharing Project. This article is intended for Image Sharing Project developers. There are several CTP articles on the RSNA MIRC wiki that might be useful references when reading this article.

The Image Sharing Project is separated into two main use cases, one for the transfer of images for clinical use and one for transfer of anonymized research data. Each use case has a sender and a receiver.

- The clinical sender is completely contained in the Edge Server and does not involve CTP.
- The clinical receiver has two alternative configurations, one using both the Edge Server and CTP and the other using CTP stand-alone.
- The research sender uses both the Edge Server and CTP together.
- The research receiver uses CTP stand-alone.

The configuration of a CTP instance is specified by an XML file called `config.xml`. See CTP-The RSNA Clinical Trial Processor for detailed information about pipelines, pipeline stages, plugins, and the general processing model of the program.

This article presents a combined configuration file example including both of the research use cases and the clinical receiver, each implemented in its own pipeline. The configuration can be modified to remove pipelines that are not required in a specific installation.

In the configuration shown below, the server is put on port 1080, and SSL is not enabled for access to the server itself, although communication with the Clearinghouse is always secure.

1 The Configuration File

This configuration example includes pipelines for the clinical receiver, the research sender and the research receiver:

```
<Configuration log="yes">
  <Server port="1080"
    ssl="no"
    keystore="keystore.jks"
    keystorePassword="edge1234"
    truststore="truststore.jks"
    truststorePassword="edge1234"/>
  <Plugin
    name="ISN AuditLog"
    id="AuditLog"
    class="org.rsna.ctp.stdplugins.AuditLog"
    root="roots/ISN/AuditLog"/>
  <Pipeline name="Research Sender">
    <DicomImportService
      calledAETTag="00120010"
      class="org.rsna.ctp.stdstages.DicomImportService"
      logConnections="rejected"
      name="DicomImportService"
```

```

    port="1081"
    quarantine="quarantines/Sender/DicomImportService"
    root="roots/Sender/DicomImportService"
    servletContext="export" />
<ObjectCache
    name="ObjectCache-RS"
    class="org.rsna.ctp.stdstages.ObjectCache"
    id="ObjectCache-RS"
    root="roots/Sender/ObjectCache-RS" />
<DicomAnonymizer
    class="org.rsna.ctp.stdstages.DicomAnonymizer"
    lookupTable="scripts/lookup-table.properties"
    name="DicomAnonymizer"
    quarantine="quarantines/Sender/DicomAnonymizer"
    root="roots/Sender/DicomAnonymizer"
    script="scripts/Sender.script"/>
<CachingXDSEExportService
    servletContext="xds-export"
    name="CachingXDSEExportService-RS"
    class="org.rsna.isn.ctp.xds.sender.CachingXDSEExportService"
    root="roots/Sender/CachingXDSEExportService-RS"
    objectCacheID="ObjectCache-RS"
    minAge="300"
    timeDepth="30"
    iti8Pix="mllps://clearinghouse.lifeimage.com:8888"
    iti8Reg="mllps://clearinghouse.lifeimage.com:8890"
    iti41="https://clearinghouse.lifeimage.com/services/xdsrepositoryb"
    iti41SrcId="1.3.6.1.4.1.19376.2.840.1.1.2.1"
    timeout="120000"
    auditLogID="AuditLog"
    quarantine="quarantines/Sender/CachingXDSEExportService-RS">

    <Destination
        name="Trial S"
        key="9652fef890fe9b5de17f2116e971f153d7d655c3b57fef6012a2138239e34d20"
        token="fpzddg"
        dob="19460201"
        pw="quackquack"/>
    <Destination
        name="Trial T"
        key="e03c75ab7056c94335e3f06a6e81331a973413efc3474ff7a508ce842a3630ea"
        token="63x4ec"
        dob="19460201"
        pw="quackquack"/>

</CachingXDSEExportService>
</Pipeline>
<Pipeline name="Research Receiver">
    <PollingXDSImportService
        name="PollingXDSImportService-RR"
        class="org.rsna.isn.ctp.xds.receiver.PollingXDSImportService"
        root="roots/ResearchReceiver/PollingXDSImportService"
        interval="60"
        rad69URL="https://clearinghouse.lifeimage.com/ImagingDocumentSource_Service?wsdl"
        registryURL="https://clearinghouse.lifeimage.com/services/xdsregistryb"
        repositoryURL="https://clearinghouse.lifeimage.com/services/xdsrepositoryb"
        repositoryUniqueID="rsna.domain.repository"
        assigningAuthorityUniversalID="1.3.6.1.4.1.19376.2.840.1.1.1.1"
        assigningAuthorityUniversalIDType="ISO"
        homeCommunityID="rsna.domain"
        siteID="9652fef890fe9b5de17f2116e971f153d7d655c3b57fef6012a2138239e34d20"
        imagesPerRequest="100"
        timeout="120000" />
    <ObjectLogger
        name="ObjectLogger-RR"

```

```

        class="org.rsna.ctp.stdstages.ObjectLogger"
        interval="1"
        verbose="yes" />
<FileStorageService
    class="org.rsna.ctp.stdstages.FileStorageService"
    name="FileStorageService-RR"
    port="1086"
    quarantine="quarantines/ResearchReceiver/FileStorageService"
    root="roots/ResearchReceiver/FileStorageService"/>
<!--
    <DicomExportService
        name="DicomExportService-RR"
        class="org.rsna.ctp.stdstages.DicomExportService"
        root="roots/ResearchReceiver/DicomExportService"
        quarantine="quarantines/ResearchReceiver/DicomExportService"
        url="dicom://DestinationAET:ThisAET@ipaddress:port"
        interval="5000" />
-->
</Pipeline>
<Pipeline name="Clinical Receiver">
    <XDSImportService
        name="XDSImportService-CR"
        class="org.rsna.isn.ctp.xds.receiver.XDSImportService"
        root="roots/ClinicalReceiver/XDSImportService-CR"
        interval="60"
        rad69URL="https://clearinghouse.lifeimage.com/ImagingDocumentSource_Service?wsdl"
        registryURL="https://clearinghouse.lifeimage.com/services/xdsregistryb"
        repositoryURL="https://clearinghouse.lifeimage.com/services/xdsrepositoryb"
        repositoryUniqueID="rsna.domain.repository"
        assigningAuthorityUniversalID="1.3.6.1.4.1.19376.2.840.1.1.1.1"
        assigningAuthorityUniversalIDType="ISO"
        homeCommunityID="rsna.domain</HomeCommunityId"
        imagesPerRequest="100"
        timeout="120000"
        servletContext="xds-import" />
    <ObjectLogger
        name="ObjectLogger-CR"
        class="org.rsna.ctp.stdstages.ObjectLogger"
        interval="1"
        verbose="yes" />
    <FileStorageService
        class="org.rsna.ctp.stdstages.FileStorageService"
        name="FileStorageService"
        port="1087"
        quarantine="quarantines/ClinicalReceiver/FileStorageService"
        root="roots/ClinicalReceiver/FileStorageService"/>
<!--
    <DicomExportService
        name="DicomExportService-CR"
        class="org.rsna.ctp.stdstages.DicomExportService"
        root="roots/ClinicalReceiver/DicomExportService"
        quarantine="quarantines/ClinicalReceiver/DicomExportService"
        url="dicom://DestinationAET:ThisAET@ipaddress:port"
        interval="5000" />
-->
</Pipeline>
</Configuration>

```

2 The Server Element

The **server** element specifies the port of the web server. It also allows the specification of the keystore and truststore files. In the Image Sharing Network (ISN), these are special files and their names and passwords must be

supplied. The `ISN-installer.jar` program installs the `keystore.jks` and `keystore.jks` files in the `CTP` directory. If other keystore or truststore files are to be used, be sure to specify their locations in the `server` element.

3 The AuditLog Plugin

The AuditLog plugin captures log entries made by the `CachingXDSEExportService` stage in the Research Sender pipeline. An entry is made at the completion of each study upload to the clearinghouse. The format of the entry is shown below:

```
2013-01-26 13:07:13.069
Content Type: xml
Entry ID: 1

<XDStudyCache
  Action="send"
  Destination="9652fef890fe9b5de17f2116e971f153d7d655c3b57fef6012a2138239e34d20"
  DestinationName="Trial S">
  <Study
    bodypart=""
    description="MR"
    destination="9652fef890fe9b5de17f2116e971f153d7d655c3b57fef6012a2138239e34d20"
    destinationName="Trial S"
    modality="MR"
    objectsSent="13"
    patientID="5400001"
    patientName="DBW-WHIMS-54"
    size="13"
    status="SUCCESS"
    studyDate="2005.12.25"
    studyDescription=""
    studyUID="2.16.840.1.113662.4.4142577143398.1106696834.2813456275582390951"/>
  </Study>
</XDStudyCache>
```

The AuditLog installs a servlet with the context `AuditLog`, providing a user with the `admin` privilege the ability to search the log and display entries.

4 The Research Sender Pipeline

The Research Sender Pipeline configuration assumes that unanonymized DICOM data objects are transmitted to CTP by an external application using the DICOM protocol. The objects are anonymized, removing PHI from the text elements. The objects are then cached until a user manually selects the study and assigns a destination, after which they are exported to the clearing house.

In this configuration, all the stages except the `CachingXDSEExportService` are standard CTP stages. They are all documented in CTP-The RSNA Clinical Trial Processor.

Note: The `DicomDecompressor` and `DicomPixelAnonymizer` stages can be added to this pipeline if PHI is burned into the pixels of images received for transmission. If the `DicomPixelAnonymizer` is in the pipeline, the `DicomDecompressor` must also be present, and it must precede the `DicomPixelAnonymizer`.

4.1 CachingXDSEExportService

The special attributes of the CachingXDSEExportService are:

- **root** is a directory within which the CachingXDSEExportService caches objects and stores its indexing information.
- **cacheID** specifies the ID of an ObjectCache stage that may be accessed by the CachingXDSEExportService to obtain pre-anonymization information so it can index the objects by their PHI and make that information available to a user who accesses the servlet. If the cacheID attribute is not supplied, the CachingXDSEExportService indexes the objects by the anonymized values.
- **servletContext** specifies the context of a servlet that provides a user interface allowing a user to select cached studies and assign them destinations for export. The purpose of the servletContext attribute is to allow each CachingXDSEExportService to have its own servlet in cases where multiple export pipelines are present in the same CTP instance.
- **minAge** is the minimum age in seconds which defines the completion of a study. When the last object received for a study is at least minAge old, the study is considered complete and is available for selection for export. The default value is **300**; the minimum accepted value is **60**.
- **timeDepth** specifies the length of time in days that studies are stored. The default value is **0**, which means forever. If timeDepth is greater than zero, studies older than that value are automatically removed from the cache, making them no longer available for export.
- **auditLogID** specifies the ID of the AuditLog plugin to be used to log studies that have been transmitted. To disable audit logging, remove this attribute.

4.1.1 The Destination Element

The CachingXDSEExportService must have at least one Destination child element specifying both a name and key for the destination on the Clearinghouse.

4.1.2 The CachingXDSEExportService Servlet

The servlet provided by the CachingXDSEExportService is installed on the CTP servlet container at the context specified by the servletContext attribute. The servlet provides the following APIs:

- **/ {servletContext}** - GET returns an HTML page containing a form for selection of studies and destinations. The page also lists the studies that have already been transmitted and shows their status (number of images transmitted, total size of the study, success or failure of the transmission).
- **/ {servletContext}** - POST accepts a completed form and returns a new HTML page.

The servlet displays studies that are available for transmission. Each study has a status that is either OPEN or COMPLETE. A study is open is one for which an object has been received in the last number of seconds specified by the **minAge** attribute.

A study can be selected for transmission whether its status is OPEN or COMPLETE. Only those images that have been received by the time transmission starts will be included in the submission set.

When a study has been selected for transmission, it moves from the studies table to the Sent Studies table on the page. Once a study has been successfully transmitted to the selected destination, it remains visible in the Sent Studies table for one hour, after which the study is deleted from the table and from storage.

The servlet automatically provides an additional destination, **Trash** to allow studies to be deleted from the cached studies table and from storage.

5 The Research Receiver Pipeline

The Research Receiver Pipeline polls the Clearinghouse for submission sets, imports them, and passes the objects down the pipe. In the configuration shown above, the same key is used for both sending and receiving, so the objects that are transmitted will be very quickly be retrieved. This is just for testing; in a real installation, the configurations of the destination keys and siteID keys will be different.

5.1 PollingXDSImportService

The special attributes of the PollingXDSImportService are:

- **root** is a directory within which the PollingXDSImportService stores its import queue.
- **interval** specifies the time in seconds between polls of the Clearinghouse. The default value is **60**. The minimum accepted value is **60**.
- **siteID** specifies the key or keys on which to poll the Clearinghouse. See the notes below for details.

Notes:

- The siteID attribute can contain multiple keys, separated by whitespace, as in:

```
siteID="9652fef890fe9b5de17f2116e971f153d7d655c3b57fef6012a2138239e34d20
e03c75ab7056c94335e3f06a6e81331a973413efc3474ff7a508ce842a3630ea"
```

- The PollingXDSImportService polls on the siteID keys, importing all the submission sets that are available. It then sleeps for the specified interval and then polls again. When multiple siteID keys are supplied, each key is polled during every interval.
- The PollingXDSImportService tracks which studies it has imported, and it does not import the same study multiple times.

5.2 ObjectLogger

The ObjectLogger stage is included in the configuration just for testing. It logs information about every object that flows down the pipe to confirm what has been received. This stage should be removed in a operational installation.

5.3 FileStorageService

The FileStorageService stage is included for testing. It saves all the objects that are received. This stage may be useful in an operational installation as well. It is configured with a port (1086) on which it opens a web server to provide browser access to the studies received.

6 The Clinical Receiver Pipeline

The Clinical Receiver Pipeline provides a user interface for selecting studies to be downloaded from the

Clearinghouse.

6.1 XDSImportService

The special attributes of the XDSImportService are:

- **root** is a directory within which the XDSImportService stores its import queue.
- **interval** specifies the time in seconds between polls of the Clearinghouse. The default value is **60**. The minimum accepted value is **60**.
- **servletContext** specifies the context of a servlet that provides a user interface allowing a user to manually enter a key to import a submission set.

Notes:

- The purpose of the servletContext attribute is to allow each XDSImportService to have its own servlet in cases where multiple import pipelines are present in the same CTP instance.

6.2 ObjectLogger

The ObjectLogger stage is included in the configuration just for testing. It logs information about every object that flows down the pipe to confirm what has been received. This stage should be removed in a operational installation.

6.3 FileStorageService

The FileStorageService stage is included for testing. It saves all the objects that are received. This stage may be useful in an operational installation as well. It is configured with a port (1087) on which it opens a web server to provide browser access to the studies received.

7 The ISN Tool Servlet

The system provides a useful servlet for creating destination keys. The servlet is accessed with the context `/isn-tool`. This servlet is installed automatically whenever a CachingXDSExportService or a PollingXDSImportService appears in the configuration.

8 Installation and Configuration

ISN is implemented as a set of CTP pipeline stages and servlets. The ISN-installer.jar (<http://mirc.rsna.org/download/ISN-installer.jar>) program contains the installer for both CTP and the ISN stages and servlets. The installation procedure is identical to that of CTP. The CTP installation instructions are on the RSNA MIRC Wiki.

The installer can also be used to upgrade an existing installation. When upgrading, the installer will not overwrite an existing config.xml file. It will also not modify any of the databases used by the program to track studies that have previously been downloaded.

After installation, the `CTP/config.xml` file must be edited to set the configuration for the site.

The Edge Server expects the CTP server to be on port 1080. This can be set during initial installation or by editing the `config.cml` file directly.

After starting CTP (see Running CTP, it is wise to check the log (`CTP/logs/ctp.log`) to ensure that the configuration doesn't conflict with other processes on the computer. The most common conflict is over a port. The ports in use in the default configuration are:

- 1080: the CTP web server
- 1081: the research sender `DicomImportService`
- 1086: the research receiver `FileStorageService`
- 1087: the clinical receiver `FileStorageService`

The log can be viewed on the Log tab of the Launcher application or on a browser by going to the `/logs` path on the server.

CTP can be installed as a Windows service as described in Running CTP as a Windows Service.

8.1 Selecting the Desired Pipelines

The standard configuration includes three pipelines. Those not required for the site should be deleted.

8.2 Configuring the Research Receiver Pipeline

For sites that require the Research Receiver pipeline, the configuration of the pipeline may need to be modified if downloaded images are to be sent to another system. The initial configuration includes a `DicomExportService`, but the element is commented out. If a `DicomImportService` is required, remove the comment tags and set the correct values in the `url` attribute.

The `siteID` attribute of the `PollingXDSImportService` element must be configured to specify the keys to be monitored. As noted above, multiple keys can be specified, separated by whitespace.

The initial configuration includes a `FileStorageService` pipeline stage to store copies of all the files that are received. In operation, this stage can be retained to provide an archive, or it can be removed if files are being routed to external systems. The `FileStorageService` stage stores images, reports, and KOS objects, organized by `StudyInstanceUID`. The `FileStorageService` stage is configured with a port (1086) that provides browser access to all the studies and images.

In operation, it will probably be desirable to remove the `ObjectLogger` pipeline stage. It is included only to assist in initial testing.

8.3 Configuring the Clinical Receiver Pipeline

For sites that require the Clinical Receiver pipeline, the configuration of the pipeline may need to be modified if downloaded images are to be sent to another system. The initial configuration includes a `DicomExportService`, but the element is commented out. If a `DicomImportService` is required, remove the comment tags and set the correct values in the `url` attribute.

All users who access the Clinical Receiver servlet must have accounts on CTP that duplicate their credentials on the

Edge Server. The accounts must be granted the **import** privilege. An admin user can create accounts and assign privileges through the User Manager. A button linking to the User Manager is available to admin users on the CTP main page.

The initial configuration includes a FileStorageService pipeline stage to store copies of all the files that are received. In operation, this stage can be retained to provide an archive, or it can be removed if files are being routed to external systems. The FileStorageService stage stores images, reports, and KOS objects, organized by StudyInstanceUID. The FileStorageService stage is configured with a port (1087) that provides browser access to all the studies and images.

In operation, it will probably be desirable to remove the ObjectLogger pipeline stage. It is included only to assist in initial testing.

8.4 Configuring the Research Sender Pipeline

For sites that require the Research Sender pipeline, **Destination** child elements must be created for the **CachingXDSEExportService** element to specify the keys of selectable destinations. In a **Destination** child element, only the **name** and **key** attributes are required. In the example configuration, the **token**, **dob**, and **pw** attributes are included only as a way of documenting how the value of the **Destination** attribute was constructed.

All users who access the Research Sender servlet must have accounts on CTP that duplicate their credentials on the Edge Server. The accounts must be granted the **export** privilege. An admin user can create accounts and assign privileges through the User Manager. A button linking to the User Manager is available to admin users on the CTP main page.

9 Notes on Upgrading CTP/ISN

9.1 Upgrading an Installation in Place

The IS-installer program upgrades all the software and standard data files, but it preserves the configuration file (**CTP/config.xml**), the home page (**CTP/ROOT/index.html**), and any anonymization files and databases defined by the configuration. Thus, an upgrade can be done in place without any loss of data.

In situations where the upgrade requires a new configuration file or home page, these files must be deleted before running the ISN-installer program in order get it to install the new ones.

9.2 Moving or Cloning an Installation

The standard configuration keeps all its software and data within the directory tree under the **CTP** directory.

To move an existing CTP/ISN to another location, it is only necessary to copy the **CTP** directory.

If the CTP instance is running as a service, however, the service must be uninstalled from the old location and reinstalled in the new one.

- Windows:
 - The files for a Windows service installation are located in the **CTP/windows** directory.

- The installation file must be modified to point to the new location.
- The ISN-installer program automatically creates the installation file during an installation or upgrade.
- See Running CTP as a Windows Service for details on installing and uninstalling the service.
- Linux:
 - The files for a Linux service installation are located in the **CTP/linux** directory.
 - The installation file must be modified manually. No special support is provided in the ISN-installer program.
 - See Running CTP as a Linux Service for instructions.

10 Notes for V&V Testing

10.1 Controlling the Logger

The **log4j properties** file installed in the program is located in the CTP directory. The contents are:

```
# Logging Configuration

# set root logger level to ERROR and give it an appender
log4j.rootLogger = ERROR, RootAppender

# make the appender
log4j.appender.RootAppender = org.apache.log4j.DailyRollingFileAppender

# give it a layout
log4j.appender.RootAppender.layout = org.apache.log4j.PatternLayout
log4j.appender.RootAppender.layout.ConversionPattern = %d{HH:mm:ss} %-5p [%c{1}] %m%n

# set the filename
log4j.appender.RootAppender.File = logs/ctp.log

# set the datepattern
log4j.appender.RootAppender.DatePattern = '.'yyyy-MM-dd

#-----
# set the starting logger levels
#-----
log4j.logger.gov = WARN
log4j.logger.edu = WARN
log4j.logger.org = WARN
log4j.logger.org.rsna = INFO
log4j.logger.mirc = INFO
log4j.logger.org.openhealthtools.ihe.atna.auditor = OFF
log4j.logger.org.rsna.isn.ctp.xds.receiver = WARN
log4j.logger.org.rsna.isn.ctp.xds.sender = WARN
```

Either or both of the last two lines can be set to DEBUG to enable logging of all the polling and transmission activities.

The file is overwritten during an upgrade.

10.2 Resetting the System

The configuration puts the root directories of all the pipeline stages under the CTP/roots directory. (Note: Just to be clear, the **CTP/ROOT** directory is the root of the server; the **CTP/roots** directory is the root of the pipeline stages.)

Each pipeline stage confines its persistent storage to its root directory.

To reset the system, stop CTP and delete the **CTP/roots** directory. When CTP starts again, it will recreate the directory, and all the pipeline stages will create their root directories within it. This will have the effect of erasing all the history that indicates which studies have been downloaded. If the configuration has a **PollingXDSImportService**, it will then download everything on the Clearinghouse for the **siteID** keys that are in its configuration.

10.3 Clearing a Single **PollingXDSImportService**

A **PollingXDSImportService** tracks studies that it has downloaded in a database contained in the **database** subdirectory of its root directory. The database consists of two files:

- **docsetIDs.db**
- **docsetIDs.lg**

To remove the download history, these files can be deleted. They will be recreated when they are required.

10.4 Controlling **ObjectLogger** Verbosity

During testing, it may be desirable to limit the amount of information logged by the **ObjectLogger** stages. To do so, set the **verbose** attribute to **no**.

Retrieved from "http://mircwiki.rsna.org/index.php?title=CTP_Configuration_for_the_Image_Sharing_Project"

- This page was last modified on 27 January 2013, at 17:20.