

# Image Session Upload Methods in XNAT

Uploading and storing image sessions is one of the core activities of almost any XNAT installation. The list of imaging types which XNAT supports is continually expanding. Currently DICOM, Analyze, ECAT, IMA, and NRD are supported in varying degrees. XNAT natively supports several modalities, including MRI, PET, and CT, and there is emerging support for combined image session types such as PET-MR. Other imaging modalities can be supported by adding data types.

This documentation will explain how images are uploaded and stored in XNAT, and give an overview of the number of methods that can be used to upload image sessions to your XNAT archive.

## How Image Sessions Are Archived



The most common supported image session format in XNAT is DICOM, so we will use that for our example in this documentation.

## Upload to XNAT - Creating the Session

An image session is composed of a series of DICOM scan files that contain two components: the image data, and image metadata. XNAT performs a series of operations to take these loosely aggregated files and stitch them together into an image session object.

As scan files are transmitting to XNAT, XNAT "listens" for this transmission to end before beginning to build the session object. This is because a PACS system will not necessarily communicate to XNAT how many files will be in a session; it just starts sending, and XNAT has to play catch. XNAT waits for a predetermined amount of time before it decides that the transmission is complete before building the session. Site-wide anonymization occurs at this point. (XNAT Admins can change this setting: [See Site-wide Session Upload and Anonymization Settings.](#))

XNAT reads the DICOM metadata in the header of each of the scan files, and looks for fields that specify the project, subject, session date and session label. These tell XNAT how to bundle and store the session object. [See How XNAT Scans DICOM to Map to Project/Subject/Session.](#) The metadata will be stored as XML in the PostgreSQL XNAT database, which allows users to search and find multiple image sessions matching certain criteria. Project-specific anonymization is applied to the scans as they are saved to the archive.

The scan files themselves will be stored in the XNAT file system, which users can download from. Each scan also has an XML representation in the XNAT database, which stores its scan type, description and scan parameters. However, these are not searchable in XNAT.

The precise order of processing steps varies with upload method, site and project settings, and the scan data. [See Image Upload Processing Steps](#) for more detail.

## Anonymization

When importing clinical scans, DICOM metadata will often contain personally identifying information about the patient. Unless your XNAT is customized to be HIPAA- and IRB-compliant for working on clinical studies, it is your responsibility to remove this PII and anonymize your image data.

XNAT offers native support for removing PII from DICOM metadata. This can be done by writing a custom anonymization script using the [DicomEdit](#) format. These scripts can be set to be run site-wide, or customized on a project-by-project basis.

- **XNAT Admins:** [Site-wide Session Upload and Anonymization Settings](#)
- **XNAT Project Owners:** [Project Data Import and Anonymization Settings](#)

It is also possible for PII such as a patient name or patient code to be burned into the image file itself. XNAT does not offer native support for this yet, but there are pipelines in development that can identify scans with this "burned-in" PII and allow you to edit or remove them.

### [Where Anonymization Happens in XNAT](#)

### [How to Write an Anonymization Script](#)

## Prearchive or Quarantine



If you suspect that your incoming scan data may contain PII in any form, we strongly encourage you to send your scans to the Prearchive.

The XNAT Prearchive lives in a separate database and file system from your XNAT Archive, and does not permanently store any data. This makes it a safe place to triage scans that may contain PII and prevent them from being permanently stored in your Archive. When scans are sent to the Prearchive, you can review them to ensure that they have been properly anonymized, and that they are assigned to the proper project and subject, before moving them to the Archive.

XNAT can be configured on a project-by-project basis whether to send all incoming scans to the Prearchive or not.

- [Using the Prearchive](#)
- **XNAT Project Owners:** [Project Data Import and Anonymization Settings](#)

XNAT also supports another method of previewing data before making it widely available, called a "Quarantine." If this is enabled, image session data is archived but given restricted access permissions until a project owner has reviewed it for quality and marked it usable. Once a session has been released from quarantine, it becomes widely searchable and downloadable by other XNAT users with permissions to see your data.



**The Quarantine feature should not be used for reviewing sessions for PII.** Sessions that have been quarantined have already been permanently stored in the XNAT Archive. Even if a session is deleted from the Archive, a record of its metadata is still preserved in the project audit trail in the PostgreSQL database.

## Autorun Pipelines

Pipeline processing can be set to run automatically when a session is moved into the XNAT archive. An example of this might be to run a protocol check to ensure that the session contains the number and type of scans that your project expects, or to run a quality check to look for common imaging artifacts.

- **XNAT Admins:** [Installing Pipelines in XNAT](#)

## Archiving a Session

MR Session: UCV002_v01_mr			
Details		Projects	
<b>Accession #:</b>	XNAT_E00016	<b>Subject:</b>	UCV002
<b>Date Added:</b>	12/12/2016 17:23:05 (admin)	<b>Gender:</b>	Female
<b>Date:</b>	07/07/2011	<b>Handedness:</b>	Right
<b>Time:</b>	13:33:07	<b>Age:</b>	64.00
<b>Scanner:</b>	Philips Medical Systems Intera		
<b>Acquisition Site:</b>	UCV		

When an image session is archived, multiple things happen:

- The session's metadata is stored in the XNAT PostgreSQL database under the specified project and subject. The project assignment will control who has permissions to view, edit, download or delete this session data.
- The session's scan files are moved into the file system and are represented in an XML file catalog. This catalog is queried to establish file counts, and to retrieve stored files for downloads.
- In addition to the session label, which is the common name that will show up in searches and project listings, XNAT assigns a pair of unique identifiers to the session object in the database. There is a private UID that is only used for internal system reference, and there is a user-accessible ID known as an **Accession ID**. The Accession ID can be used in API queries.

As mentioned above, all of the information that is stored in the XNAT PostgreSQL database is permanently archived. This means that even if a session is marked for deletion, this merely updates the record in the database. Data is not actually destroyed or removed by any XNAT operation.

## Methods for Uploading Sessions

### Importing from a DICOM AE or PACS to XNAT

XNAT natively supports connections to DICOM scanners, PACS systems, and other image session routing systems. These connections must be set up by an XNAT administrator. When these connections are active, XNAT users are not required to perform any action in the UI to initiate a scan import. Instead, scans can be sent directly from the scanner or PACS system to your XNAT. XNAT will always be "listening" for uploads to begin, and will begin processing them as they arrive.

- **XNAT Admins:** [Connecting XNAT to DICOM Scanners and PACS](#)

## XNAT Upload Applet



Many browsers including Chrome and Safari have ended or will end support for embedded Java applications, signaling an upcoming end of life for the Java-based XNAT Upload Applet. We recommend using Firefox in the short term, or exploring other upload options listed below.

The XNAT Upload Applet is an embedded Java applet that can run inside your web browser and allow you to browse your computer's file system and connected drives for an image session object. The applet provides a step-by-step wizard for uploading your data and assigning it to the appropriate project and subject.

**Note:** The XNAT Upload Applet can be configured to require the user to know certain scan parameters, such as the scan date, before allowing an upload to begin. XNAT Admins can set this setting.

- **XNAT Users:** [Using the Upload Applet](#)
- **XNAT Admins:** [Site-wide Session Upload and Anonymization Settings](#)

## XNAT Desktop Upload Assistant

The XNAT Upload Assistant is essentially a copy of the XNAT Upload Applet that is installed on your desktop, and does not require a web browser or Java support. It offers the same step-by-step wizard for uploading data and assigning it to the appropriate project and subject. The XNAT Upload Assistant can be run on Mac OS X, Windows or Linux, and requires no special setup by an XNAT administrator.

- **XNAT Users:** [Using the Desktop Upload Assistant](#)

## Compressed Image Uploader

The Compressed Image Uploader is an alternative that runs in the XNAT browser and does not require a local installed application. The compressed image uploader requires an image session that has been compressed using .zip or .tar.gz format. It is useful for uploading image sessions one at a time.

- **XNAT Users:** [Using the Compressed Image Uploader](#)

## Uploading via XNAT API

If you have large numbers of sessions to archive, you may consider writing a script to manage your uploads. Such a script would make use of the XNAT API to connect to your XNAT instance and perform the upload and archive functions.

- **XNAT Developers:** [Upload Image Session Files with REST API](#)

## Uploading via DICOM Inbox

Another method of uploading large numbers of sessions involves moving your session files to a mounted file system that your XNAT server can access, then pointing the XNAT importer to it with a REST call. This bypasses the HTTP protocol during the import process, which can speed things up.

- **XNAT Sys Admins:** [Using DICOM Inbox to import an image session](#)