

# XGet Manual

## Quick Index

- [1 Introduction](#)
- [2 System Requirements](#)
- [3 Download](#)
- [4 Installation](#)
- [5 Command-Line Flags](#)
- [6 Authentication + Host Name](#)
- [7 Specifying Sessions](#)
- [8 Specifying The Output Directory](#)
- [9 Downloaded Files](#)
- [10 Adding Constraints](#)
  - [10.1 Scans](#)
  - [10.2 Assessors](#)
  - [10.3 Reconstructions](#)
  - [10.4 Project](#)
  - [10.5 Format](#)
- [11 XML Field Mapping](#)
- [12 Future Enhancements](#)

## 1 Introduction

*xget* is a command-line tool written in Python that downloads images from an XNAT server. It is more convenient than the REST API because it allows the user to place more granular constraints on what images they would like to download. It uses the *pyxnat* library to interact with the XNAT server.

## 2 System Requirements

Python >= 2.6 Linux/Windows

## 3 Download

Operating System	Bundle
Linux	<a href="http://ftp.nrg.wustl.edu/pub/tools/xget">http://ftp.nrg.wustl.edu/pub/tools/xget</a>
Windows	<a href="http://ftp.nrg.wustl.edu/pub/tools/xget.exe">http://ftp.nrg.wustl.edu/pub/tools/xget.exe</a>

The source is located in a Bitbucket repository at : <http://bitbucket.org/deech/xget-public>.

To get it, install [Mercurial](#) and run:

```
hg clone http://bitbucket.org/deech/xget-public
```

## 4 Installation

Install the packages listed under the "Prerequisites" section of <http://packages.python.org/pyxnat/installing.html>. There is no need to install *pyxnat-httplib2* or *pyxnat* itself.

## 5 Command-Line Flags

Flag	Description
-h	Print a help message
-u	username, see <a href="#">Authentication + Host Name</a>
-p	password, see <a href="#">Authentication + Host Name</a>
-host	hostname, see <a href="#">Authentication + Host Name</a>
-passfile	passfile path, see <a href="#">Authentication + Host Name</a>
-s	Session id/label(s) of desired session(s). See <a href="#">Sessions</a> .
-proj	Retrieves only sessions in this project. See <a href="#">Project</a> .
-start	Retrieve experiments created on and AFTER this date. Uses DAYMONTHYEAR format eg. 01012011 is January, 1st 2010.

-end	Retrieve experiments created on and BEFORE thisdate. Uses DAYMONTHYEAR format eg. 01012011 isJanuary, 1st 2010.
-acq	Retrieves scan(s) of the specified type ordescription. See <a href="#">Scans</a> .
-q	Quality of scans. Defaults to ALL. See <a href="#">Scans</a> .
-ass	Retrieve assessors for the specified sessions. See <a href="#">Assessors</a> .
-recon	Retrieve reconstructions for the specified sessions.See <a href="#">Reconstructions</a> .
-format	Image format, defaults to DICOM
-o	Output directory. See <a href="#">Output Directory</a>
-z	Extract the downloaded zips. Defaults to FALSE. True if this flag is present. See <a href="#">Downloaded Files</a> .
-overwrite	If present overwrite an existing zip file, defaults to false. See <a href="#">Downloaded Files</a> .
-longname	Capture more constraints in the name of the downloaded ZIP archive. See <a href="#">Downloaded Files</a> .

## 6 Authentication + Host Name

You may specify your username, password and hostname:

- On the command-line:

```
./xget -u admin -p admin -host http://localhost:8080/xnat
```

- In an XNAT pass file. If any of the hostname, username or password are missing from the command-line, *xget* will look in this file to fill in the information. The XNAT pass file is usually stored at the root of the user's home directory and called `.xnatPass` (`~/.xnatPass`). The format for entries in the `xnat` is:

```
+username@www.xnathost.org=password
```

where "username" is your XNAT username, "password" is your XNAT password and "www.xnathost.org" is your XNAT's host name.

An example XNAT pass file might look like this:

```
+admin@http://localhost:8080/xnat=admin
```

If the pass file is stored elsewhere that path must be specified on the command-line:

```
./xget -passfile /path/to/passfile
```

## 7 Specifying Sessions

*xget* understand both the session label and the session id. A session or a list of session can be specified either:

- On the command line, for example:

```
./xget ... -s xnat_E0001,xnat_E0002,sess_1
```

Note that you can mix session labels with ids. Above we have told *xget* that we are interested in the sessions with ids "xnat\_E0001, xnat\_E0002" and label "sess\_1".

- And/or in a file where each line a list of sessions, for example the following is equivalent to the command-line argument above:

```
xnat_E0001,xnat_E0002
sess1
```

## 8 Specifying The Output Directory

The output directory is the directory into which files are downloaded, for example:

```
./xget -o tmp
```

will download images (or zip files) into a "tmp" folder in the current directory.

**NOTE:** *xget* expects this directory to exist and the user running it to have write permissions to it.

## 9 Downloaded Files

By default image files are downloaded as ZIP archives and placed in the directory specified on the [command-line](#), and have a default name of:

```
session-label-or-id_resource-type.zip
```

For example if *xget* is told to download all images in the session "sess\_1" into the "/tmp" directory once downloading is complete the following 3 files will be available:

```
/tmp/sess_1_acq.zip -- the scans associated with sess_1
/tmp/sess_1_ass.zip -- the assessors associated with sess_1
/tmp/sess_1_recon.zip -- the reconstructions associated with sess_1
```

It is also possible to tell *xget* to name the zip files so that more download constraints are captured. The filename format will now look like the following:

```
p_project-name_sess_session-label-or-id_resource-type_constraints_q_quality_f_format.zip
```

For example,

```
./xget ... -s sess_1 -proj proj_3 -longname ...
```

tells *xget* to download all the files from the "sess\_1" experiment which is in the project "proj\_3". After downloading the following three ZIP files will be available:

```
/tmp/p_proj_3_sess_sess_1_acq_ALL_q_ALL_f_DICOM.zip -- the scans
/tmp/p_proj_3_sess_sess_1_ass_ALL_q_ALL_f_DICOM.zip -- the assessors
/tmp/p_proj_3_sess_sess_1_recon_ALL_q_ALL_f_DICOM.zip -- the recons
```

If the "-z" flag is set, the ZIP archives are downloaded to the [output directory](#), extracted and then deleted.

By default if a zip file exists it is not downloaded again. Setting the "-overwrite" flag overrides this behavior and replaces the existing ZIP archive on your filesystem.

**NOTE:** Currently the "-overwrite" flag only uses the zip file name to determine if it already exists. If for example a ZIP archive is downloaded using some set of constraints and the "-longname" flag, and then another is downloaded using the same set of constraints but without the flag, it will be downloaded again even though the files within the ZIP archive are probably the same.

## 10 Adding Constraints

### 10.1 Scans

Constraints on what scans to download are specified using the "-acq" flag, which can take a comma-separated list of scan id's and scan descriptions. For example:

```
./xget ... -s sess_1 -acq 1,TSE,"PIB 336mtx" ...
```

will download scans from "sess\_1" where the scan id is "1", the scan description is "TSE" or the scan description is "PIB 336mtx". It will also download all the assessors and reconstructions. Note that scan descriptions with spaces must be quoted.

Setting the "-quality" flag will only download scans of the specified quality. For example,

```
./xget ... -s sess_1 -acq 1,TSE,"PIB 336mtx" -quality usable ...
```

will only download scans with the specified scan ids and descriptions but only if they are marked "usable".

### 10.2 Assessors

Specifying which assessors is similar to downloading [scans](#), except that the command-line flag is "-ass".

### 10.3 Reconstructions

Specifying which assessors is similar to downloading [scans](#), except that the command-line flag is "-recon".

### 10.4 Project

If a project is specified on the command-line *xget* will only look for sessions within that project, for example:

```
./xget ... -proj proj_1 ...
```

### 10.5 Format

Setting the "-format" flag will constrain download to images of that type. It currently defaults to "DICOM".

## 11 XML Field Mapping

*xget* download constraints are directly mapped to fields and attributes in that sessions' XML. The mappings are as follows:

Resource	Attribute	XPath
Session	Project	/@project
	ID	/@ID
	Label	/@label
	Date	/date
Scan	ID	/scans/scan/@ID
	Quality	/scans/quality
	Format	/scans/scan/file/@format
Assessor	ID	/assessors/assessor/@ID

	files	/assessors/assessor/out
	type	/assessors/assessor/type
Reconstructions	ID	/reconstructions/reconstructedImage/@ID
	files	/reconstructions/reconstructedImage/out
	type	/reconstructions/reconstructedImage/type

## 12 Future Enhancements

- The ability to specify the subject as a constraint
- The ability to download **all** sessions in a project. For now a (list of) session(s) is mandatory.

Author: Aditya Siram