

**What has been added to xtardis:**

## Active Directory Authentication Module:

An additional django module to provide authentication against an active directory server when creating new accounts and when logging in.

## PID Generation Module:

A module to query the ANDS PID/DOI server to request a PID which is then attached to each experiment that is made public.

## DOI Generation Module:

Bundled with the PID module is the ability to assign a DOI to experiments.

## RIF-CS Generation Module::

A module that when queried converts the xtardis relational database into RIF-CS compatible XML for integration into the RDA database.

## Metadata schemas for NanoTOF &amp; MasterSizer:

During the initial database synchronisation the metadata schemas for the NanoTOF and MasterSizer are integrated into the database by default.

**Which components of xtardis have been altered:**

## Interface:

Minor cosmetic changes to the interface were made to conform to the IWRI.

## User Accounts:

The user account system was heavily modified to contain additional information as required by the RIF-CS person specification. Along with additional levels of experiment access privileges.

## Account Creation:

Changed account creation procedure to fit with new schema.

## Schema Modifications:

Changes to other database schemas to conform to the RIF-CS specification.

## Cleaning:

Removed myTardis specific code that was not applicable to xTardis.

## Bug Fixes:

Miscellaneous bug fixes.

## **New software/scripts written for this project:**

### **Simple Install Script:**

A straightforward install script that downloads all required software, libraries, and python eggs then decompresses and installs them ready for xTardis to be configured then run.

[https://code.google.com/p/mytardis/source/browse/branches/xtardis/ubuntu\\_server\\_setup.sh](https://code.google.com/p/mytardis/source/browse/branches/xtardis/ubuntu_server_setup.sh)

### **Client Software:**

Java based client side application to parse machine output for desired meta-data and automatically upload. Currently not on a public repository after it was determined the NanoTOF used a proprietary file format which the owners were unwilling to provide the header specification for, and the Mastersizer desired metadata is actually derived from, instead of being present in, the header of their file format. Will hopefully be used in the future as more machines come online with friendlier file formats.

The client waits for an experiment to complete, then automatically extracts the metadata from the produced output files. This metadata is then associated with the logged in user and uploaded to the xTardis server along with the raw data.