# SYNGO.SHARE/BIOBANK INTEGRATION DOCUMENTATION

## Introduction

The system covered in this document contains the CTSU part of the management of DICOM data generated from the greater BioBank study, how this data is processed and stored in the BioBank archive. This includes identification of the incoming data, processing, storage and where necessary providing clinical information, where a possible serious incidental medical finding is flagged.

## Overview

The first part of the task is to collect all this data at one location, the devices used in the BioBank study which generate DICOM files already have the functionality to push that data to a central server running a PACS system. Once in this system, the DICOM is checked over by a human and keywords applied to help future research and categorisation.

After which point our processes come into play, they monitor this central system, matching up data to BioBank participants, downloading both the attached keywords and DICOM, processing and storing the files into the BioBank archive. We also detect any which have been identified as a possible serious incidental finding and which require further assessment by an expert. In these cases we extract anonymised clinical information for that participant and upload it to the PACs for further examination.

## System Process

BioBank devices that generate DICOM will export it to the BIOBANK PACS running sygo.share.

**\*All\*** DICOM modalities entering the PACS get stored in “BIOBANK\_QA”. For the first 1000 scans, a copy will also be automatically placed in “1KSCANS”.

On the MRI scanners, the resulting scans will be checked; at this point possible P.S.I could be raised. The MRI radiographer will get it reported on by the Lead Radiographer at this point, or copy it to “BIOBANK\_PSI” and bring it to the attention of the Incidental Findings Administrator. If confirmed a description will be uploaded to the scan and is keyword tagged accordingly.

In any case the scan is automatically uploaded to “BIOBANK\_QA” as per normal.

**\*ALL MODALITIES\*** in “BIOBANK\_QA” will be quality checked by a human and tagged with appropriate keyword(s) where required, before being moved to “BIOBANK”. These keywords will be used to by researchers to defined required image quality in their research. Only scans which are not at all fit for use will be removed off the system at this point. There should always be a full set of images for each participant; if there is not then this should be flagged. As on the MRI scanner, a P.S.I. maybe noticed and a copy will be sent to “BIOBANK\_PSI” for further reporting*.*

Once in “BIOBANK\_PSI”, CTSU will upload participant’s clinical information taken from the BioBank touchscreen questionnaire, this data will still anonymous.

The Incidental Findings Administrator monitors “BIOBANK\_PSI”, once the participant details are in place, it is moved to an appropriate review radiologist’s Authority Unit. The Administrator will alert the review radiologist to the new cases as required.

The review radiologist will tag *and upload* any findings and move the lot to “REPORTED”.

The Incidental Findings Administrator will take the scans in “REPORTED” and perform the appropriate action. If no referral is needed, it gets moved to “COMPLETED”. If PSI is confirmed, two letters are posted, to the participant, asking to make an appointment with their GP and to the GP, with details of the reports. Both will have details on how to request CD’s of the relevant images. Once this is done, the Incidental Findings Administrator moves the images to “COMPLETED”.

CTSU will automatically download all new scans and their attached keywords from “BIOBANK”. Prepare the data and archive it within the BioBank storage.

There are various SOPs which cover this in more detail. These can be requested from Steve Garrot. The ones I know about are:

CLI82 - Imaging Incidental Findings SOP

CLI83 - Reporting Incidental Findings for UK Biobank SOP

|  |  |
| --- | --- |
| DICOMBIOBANK  When a study is downloaded from the syngo.share, it is stored in this table. | |
| patientid | The DICOM participant id |
| studyid | Dicom Study id. Retrieved from the Dicom files |
| studyid\_enc | Encrypted study id |
| seriesid | Dicom Series id. Retrieved from the Dicom files |
| seriesid\_enc | Encrypted series id |
| studydate | The date the study was inserted into the PACS |
| statusenc | 1 if we have not decrypted the encrypted fields, 0 if decrypted. Used by the decryption cron job |
| zipmd5 | MD5 of the downloaded and zipped DICOM study |
| studypk | The Syngo.Share PK of the study |
| seriespk | The Syngo.Share PK of the series |
| orginalsize | The pre-zip size of the DICOM |
| compressedsize | The zip of the zip |
| importstatus | Set to 1 when the DICOM import table has been written to |
| status | 1 if OK, anything else is invalid field |
| numImages | The number of images in this Series |
| modality | The modality(s) of the series, see DICOM Standard for meaning |
| aet | The name of the device which preformed the scan. |
| aesvi | The VI of the AES for this row. |
| filename | Series Zip file name. Format: Modality\_trid\_cid\_SeriesPK\_StudyPK.zip |

# Databases

The system uses ace\_ice\_live, ace\_ice\_trove\_live, extract\_pacs\_live and extract\_pacs\_ddb on BB5.

The databases ace\_ice\_live and ace\_ice\_trove\_live are used to match up dicom participant id’s to biobank participant id’s, then to extract the biobank quiz answers/questions as required. Part of this process uses extract\_pacs\_ddb, a distributed database for finding out when entries have yet to be processed.

The extract\_pacs\_live database is where all the data is stored for the running of this system. It was created by user Biocore, accessed by extract.

Extract\_pacs\_live database contains the following tables:

|  |  |
| --- | --- |
| LOGS  Errors and information is stored in this table, which can be viewed using the website | |
| id | PK |
| inserttimestamp | Time stamp |
| loglevel | 0=DEBUG,1=INFO,2=ERROR,3=CRITIAL,4=ACTIONREQUIRED |
| logtype | 0=GENERAL,1=DATABASE,2=FILESYSTEM,3=PACS |
| patientid | The Dicom patient id, if applicable |
| studydate | The Study date, if applicable |
| status | 1 if valid entry |
| message | The log message |

|  |  |
| --- | --- |
| DICOMDOWNLOADER  Keeps track of the last time the PACS was scanned for new DICOM entries. | |
| Lastdownload | Timestamp |

|  |  |
| --- | --- |
| BIOBANKPARTICIPANT  As BioBank questions are submitted into ICE, a CRON job extracts the following data and places in this table. | |
| pid | The pid of the Patient |
| dicomparticipantid | DICOM Participent id for matching |
| firstnameid | DICOM FirstName id for matching |
| cid | From the ICE entery |
| trid | From the ICE entery |
| scandate | The date it was entered into ice |

|  |  |
| --- | --- |
| KEYWORD  Where details of the series level keywords attached to DICOM is stored. | |
| keywordpk | PK of the syngo.share keyword |
| keyword | The text of the keyword |
| type | Type of this keyword, taken from syngo.share |
| status | 1 if valid entry |

|  |  |  |  |
| --- | --- | --- | --- |
| DICOMBIOBANK\_KEYWORD  A table to link together which keywords belong to which series | | |  |
| seriespk |  | FK into DICOMBIOBANK |  |
| keywordpk |  | FK into KEYWORD |  |

|  |  |
| --- | --- |
| Cronstatus  A Table to keep a note of when each CRON job starts and completes. It is displayed in the web site overview. | |
| name | Name of the CRON |
| started | Timestamp of last time it started |
| finished | Timestamp of the last time it completed |

|  |  |
| --- | --- |
| Pacsstudy  Every new study found on the PACS get stored in this table. | |
| patientid | The patientid found in the DICOM file |
| firstname | The firstname found in the DICOM file |
| confirmedpatientid | Either a automatic match on either patientid or firstname into the ice system or one manually add from the web site. |
| studyuid | The DICOM study ID |
| studyuid\_enc | Encrypted version of studyid |
| studydate | The date the study was inserted into the PACS |
| aet | The name of the device which send study to the PACS |
| studypk | The syngo.share study primary key |
| modalities | The modalities syngo.share reports which are in this study |
| au | The authority unit this study is in |
| insert | Timestamp this was inserted |
| update | Timestamp this was last updated |
| status | Invalid =0, OK=1, Complete=2, Ignore = 3 |
| patientpk | The syngo.share DICOM patient primary key |
| statusenc | 1 if we have not decrypted the encrypted fields, 0 if decrypted. Used by the decryption cron job |
| aesvi | The VI of the AES for this row. |

|  |
| --- |
| STAFF |
| Name |

# THE CRONS WITH THEIR PROCESS DESCRIPTIONS

The task of keeping tack of what is going on in the syngo.share system and performing the required actions is split up into 5 processes. Each process is a Java commandline program which are independently run as a CRON job. Each CRON job will be run periodically, care has been taken so if a process task has yet to complete when the CRON kicks it off again each process can detect this and the process will safely abort.

**ScanPACS**

*overview*

This CRON fills the table PACSSTUDY, which all other CRON jobs work off.

*method*

On each run ScanPACS queries the Syngo.share for all the ‘new’ studies on the PACS in any of the following Authority Units:

* BIOBANK
* BIOBANK\_PSI
* 1kSCANS

'new’ is currently defined as ALL studies between 2 weeks previous to the last time it was scanned and now. So long as these studies StudyPK and AU isn’t registered, it is accepted as a study, which needs processing. This overlap in studies can be reduced once the system is running should it take up too many resources. Before it adds a new row, it tries to match the participant id and/or first name into the biobankparticipant table. If we get a match, then it is entered as confirmedpatientid, and can then be processed by other CRON jobs. The last scan time is kept in the table DICOMSTUDYDOWNLOADER.

*Technical details*

* It is called: cmdlinescanpacs.jar
* The CGI is installed on server: webfu@bb5
* At location: /CRONS/?
* CVS location: jonathanp/eclipse/?

**DecryptDB**

*Overview*

Decryption of all encrypted fields in a table.

*Method*

Decrypts the encrypted columns where statusenc = 1. For each table, it extracts all columns which ends with \_enc, decrypts those strings and inserts the decrypted strings into the tables columns which don’t end with \_enc. Each table which has encryption contains the column aesvi which is the VI used by the row.

*Technical details*

* It is called: cmdlinescanpacs.jar
* The CGI is installed on server: webfu@bb5
* At location: /CRONS/?
* CVS location: jonathanp/eclipse/?

**ProcessIncidents**

*Overview*

The extract of clinical information from ICE, converting it into a PDF and uploading it to the syngo.share participant in BIOBANK\_PSI.

*Method*

It retrieves all unprocessed rows from pacsstudy which are in au = ‘BIOBANK\_PSI’ and ‘1KSCANS’. If they don’t have a confirmedpatientid, it tries to match the patientid,firstname with BIOBANKPARTICIPANT’s dicomparticipantid or firstnameid. If we have a match, it uses the corresponding pid and extracts from ICE the biobank questionnaire’s clinical information and generates a PDF. The participants answers are retervied from ace\_ice\_live, the corresponding question text is reterved from ace\_trove\_live.

See Appendix B for futher details.

This PDF is uploaded to the corresponding au’s scan.

*Technical details*

* It is called: cmdlinescanpacs.jar
* The CGI is installed on server: webfu@bb5
* At location: /CRONS/?
* CVS location: jonathanp/eclipse/?

**QueryBioBankQuestionnaires**

*Overview*

Monitors ice for newly submitted biobank questionnaires and fills the biobankparticipant table so the identities of these DICOM Scans can be confirm and the clinical information can be retrieved if required.

*Method*

The CRON monitors ace\_ice\_live for any new entries which aren’t already in biobankparticipant. New enterie’s corresponding ROSETTA is decoded and the DICOMS participant id and first name is extracted. This information is then inserted into biobankparticipant.

*Technical details*

* It is called: cmdlinescanpacs.jar
* The CGI is installed on server: webfu@bb5
* At location: /CRONS/?
* CVS location: jonathanp/eclipse/?

**RetreveDicom**

*Overview*

This CRON is task is to download DICOM scans from Syngo.share. Once downloaded it generates a manifest, and compresses the data into each series the files belong too. The resulting data is submitted into the BioBank storage system for storing.

*Method*

All rows in pacsstudy with au = BIOBANK and aren’t complete (status =1) are checked to see if they have a confirmedpatientid. If not, it tries to match the patiantid and firstname into the biobankparticipant table.

For all confirmed studies it downloads each series which belongs to that study. It stores various details of the file into a manifiest CSV file for that scan. It then compresses each series with the manifest file and adds the details to dicombiobank.

It then adds an entry into the import table of the biobank system, notifing that there is data ready to be archived.

Filename of the resulting zipped DICOM series have filenames of: modality\_trid\_cid\_studyPKseriesPK.zip

*Technical details*

* It is called: cmdlinescanpacs.jar
* The CGI is installed on server: webfu@bb5
* At location: /CRONS/?
* CVS location: jonathanp/eclipse/?

# WEB SITE

We provide a basic web interface into the system; this interface provides an overview of the CRONs status, processing status and a way to manually match up scans to their ICE pids.

*Technical details*

*Web page*

* The web site is a C++, running as a CGI.
* It is called: ???.cgi
* The CGI is installed on server: webfu@bb5
* At location: /CRONS/?
* CVS location: jonathanp/?

CGI comms

Psicgi.cgi

# APPENDIX A

## Definitions

AU – Authority Unit – A ‘partition’ on the Syngo.share PACS. Each one needs authorisation to view or modify the containing data, so only those who should, can access the medical data. These are show below as “AUTHORITY\_UNIT”.

P.S.I. – Potentially Serious Incident. A possible medical issue noticed by a radiographer, radiologist or other specialist.

PACS - Picture Archiving and Communication System.

Syngo.share - The VNA software supplied by Siemens to manage the PACS.

VNA - Vendor Neutral Archive

# APPENDIX B

The clinical information on ICE\_LIVE.

Server bb5

In database ACE\_ICE\_LIVE, in the table gobz has the data from every event at a BioBank clinic stored.

Each event at the clinic is stored as a new row in the gobz table. The column Gtyp specifies which event type it refers too.

The table below details the values we are interested in retrieving from the system to build up the clinical information. To find the id number of the question text, you have to enter the trove field id into <http://bb5.ctsu.ox.ac.uk/crystal/qsearch.cgi?sa=1&sd=1>, which gives you the index into the table ace\_trove\_live.

Prefixes:

* o. value to be found in column xub\_osr
* value to be found in column xub\_isr

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Gtyp | Value | Trove field |
| Age | 11 | o.dob | calculate from date of visit (tsig on record) |
| Sex | 11 | o.male | T or F |
| List of self-reported cancer and non cancer illnesses + dates (year) of diagnosis | 31 | i.data.Vcanc.count  i.data.Vcanc.c#  i.data.Vcanc.w# | 82  84 |
| List of self-reported operations + dates (year) of diagnosis | 31 | i.data.Vnonc.count  i.data.Vnonc.c#  i.data.Vnonc.w# | 85  87 |
| *Current employment status* | 21 | i.answer.a.D9.ans.ans# | 6142 ans: [100295](http://bb5.ctsu.ox.ac.uk/crystal/coding.cgi?id=100295&sd=1&sa=1) |
| *Ethnic group* | 22 | i.answer.a.YE1.ans | 1657 |
| *Smoker – ever/never/current* | 22 | i.answer.a.S1.ans  i.answer.a.S4AA.ans | 1239  1249 |
| *Alcohol – previous or current heavy drinker* | 22 | i.answer.a.A1.ans | Aswsers:[100402](http://bb5.ctsu.ox.ac.uk/crystal/coding.cgi?id=100402&sd=1&sa=1). |
| *Systolic BP* | 31 | i.data.bp#.sys  i.data.bp#\_sys | 4080  93 |
| *Diastolic BP* | 31 | i.data.bp#.dia  i.data.bp#\_dia | 4079  94 |
| *Weight* | 34 | i.data.imped.wgt  i.data.weight | 3098  3160 |
| *Height* | 34 | i.data.hstand | 50 |
| *BMI* |  | calculate from weight & height | |