Study Closeout PDF Generation

# Introduction

The study closeout PDF generation process generates a PDF file for each patient containing all data records for that patient. Each data record is displayed as a CRF form, a data field listing and optionally, faxed in images, a chronological audit trail and an audit trail by field. Bookmarks take you to specific records and fields are hyperlinked between the CRF view, the data list view and the field audit view.

The generation process is done in two parts. The first is extracting the data and audit records from DataFax and storing them in an intermediate database. The second phase reads that database and generates the PDF files themselves.

# Generating the Intermediate Database

The process of generating the intermediate database involves running the make\_closeout\_db.py program. This program requires at least the --study argument to specify which study to generate the database for. An optional --db option allows the specification of the output database name. This program needs to be run using Python 2.7 or later.

python27 make\_closeout\_db.py --study 254 --db data.db

# Generating the PDF files

Once the intermediate database has been generated, the task of generating the output PDF files can begin. This is accomplished by running the closeout.py program which generates a PDF file for each patient containing the CRF image overlaid with the data fields, a data field listing, primary fax image, chronological audit for the record and an audit listing by field showing changes made to each field.

The closeout.py program has many options to filter and customize its output.

|  |  |
| --- | --- |
| --blinded | Blinds fields marked as internal in the DataFax setup file. |
| --db *database* | The name of the intermediate database to use. Default is ‘data.db’. |
| --studydir *path* | (Required) The path to the DataFax $STUDY\_DIR. |
| --ids *id-list* | The list of patient IDs to include. Default is to include all patients. |
| --plates *plate-list* | The list of plate numbers to include. Default is to include all plates. |
| --visits *visit-list* | The list of visit numbers to include. Default is to include all visits. |
| --domains *domain-file* | The file containing the domain to plate mapping |
| --include-attached-images list | Include primary fax images in the PDF document for the plates in ‘list’. List can be ‘ALL’. Fax images must be in PNG format. |
| --exclude-chronological-audit | Do not produce the chronological audit sections. |
| --exclude-field-audit | Do not produce the field level audit sections. |
| --prefer-background bkgd | Prefer using backgrounds with CRFtype of bkgd |
| --format-pid format | Format the patient ID according to ‘format’. Format can be any characters to include additional information such as protocol number. Any # characters in the format will be replace with a digit from the zero padded patient ID. |
| --quiet | Reduce the amount of status information displayed as PDF is generated. |
|  |  |

For example, to create a blinded PDF file for patient 99001 you would execute:

python27 closeout.py --studydir /opt/val254 --blinded --ids 99001

An output file named 99001.pdf will be created in the current directory.

# Domain File format

The domain file is used to map plate numbers to domain areas. The file format is pipe delimited with two fields – the domain name and the list of plates in that domain. For example:

Baseline|1,201,2,202,3,203,4,5-7,8,208,9-15,215,16,216,101  
Followup|17-33  
Labs|61-64  
Adverse Events|460-463,468,497,499  
Pregnancy|464-466

The domain file is used to create bookmarks by domain, visit and plate.

# Color Coding

The PDF files generated by closeout.py contain color coding. Fields marked in orange indicate missing value codes, red indicates malformed data values (values larger than the format allows), and cyan indicates data values larger than the box on the CRF can comfortably display. These cyan fields are hyperlinked to the data list view which shows the full data value.