

DataFax User Guide

Clinical DataFax Systems Inc.

Table of Contents	1
DataFax User Guide	1
Preface	2
1. Getting Help	2
2. DataFax Mailing List	2
3. Conventions	2
Chapter 1. Introduction	4
1.1. Why DataFax?	4
1.2. About This Guide	4
1.3. Basic Concepts	5
1.3.1. DataFax CRFs	5
1.3.2. DataFax Plates	5
1.3.3. DataFax Database	5
1.3.4. DataFax Keys	6
1.3.5. Record Status	7
1.3.6. Duplicate Copies of a CRF Page	7
1.3.7. Validation Levels and Work Flow Management	7
1.4. Starting DataFax	8
1.5. X11/Motif and DataFax User Interface	9
1.5.1. Using the Three Button Mouse	9
1.5.2. DataFax User and Study Toolboxes	9
1.5.3. Common Interface Components	11
1.5.4. Mnemonics	12
1.5.5. Dialog Boxes	13
1.6. Obtaining Help with DataFax	17
1.6.1. Introduction	17
1.6.2. Using Help from the DataFax User Tool Box	18
1.6.3. Using Help from the DataFax Study Tool Box	20
1.6.4. Reading DataFax documentation with a web browser	22
Chapter 2. Setup Tool	23
2.1. Introduction	23
2.2. DataFax Database	23
2.3. Getting Ready	24
2.4. Starting a New Study Setup	25
2.5. Importing CRFs	26
2.5.1. Importing a PostScript/PDF Document into DFsetup	27
2.6. Moving Between Pages	29
2.6.1. The Current Page and Total Pages Counter	29
2.6.2. Moving to the Previous and Next Pages	30
2.6.3. Go To an Arbitrary Page	30
2.7. Editing Page Definitions	30
2.7.1. Examples of Plate Triggered Procedures	32
2.8. Reordering Pages	33
2.9. Saving Your Work	33
2.9.1. The Study Properties Window	33
2.10. Quitting DFsetup	34
2.11. DataFax Variables	34
2.11.1. Basic Features	35
2.11.2. Specific Features/Coding	40
2.11.3. Edit Checks	43
2.11.4. Key Field Definition	48
2.12. Variable Styles	48
2.12.1. The Style Catalog	49
2.12.2. Creating a New Variable Style	50
2.13. Modifying an Existing Style	64
2.14. Deleting a Style	65

2.15. Defining Variables	66
2.15.1. Set DFsetup Preferences	66
2.15.2. Guides Settings	68
2.15.3. Moving Between Variables	76
2.15.4. Modifying a Variable	77
2.15.5. Deleting a Variable	78
2.15.6. Copying and Pasting Variable Definition(s)	79
2.15.7. Importing Pages and Styles from Another Study Setup	81
2.15.8. Creating Hidden Variables	83
2.15.9. Reordering Variables	84
2.15.10. Modifying the Setup of a Study in Progress	86
2.15.11. Create Menu	91
2.15.12. Meta-Words	92
2.15.13. Backups and Auto Saves	93
2.15.14. Restoring Setup from the Backup	94
2.15.15. Restoring Setup After a Program Failure	94
2.15.16. Printing a Study Schema	95
2.15.17. Testing Study Setup	96
2.15.18. DataFax Study Setup Limits	97
Chapter 3. Configuration Tool	100
3.1. Introduction	100
3.2. Starting DFconfig	100
3.2.1. DFconfig Permissions	100
3.3. DFconfig Window	101
3.3.1. Choosing a Configuration Window	103
3.3.2. Configuration Window Editors	103
3.4. Spreadsheet Editor	105
3.4.1. Common Features	105
3.4.2. Navigation	107
3.4.3. Row Actions	107
3.5. Centers Database	108
3.5.1. Error Monitor	109
3.6. Visit Map	110
3.6.1. Visit Map Editing	110
3.6.2. Cycle Definition	110
3.6.3. Visit Definition	112
3.6.4. Check Visit Map Consistency	114
3.7. Conditional Maps: Termination, Cycle, Visit, and Plate	115
3.7.1. Defining a conditional map	115
3.7.2. Defining a conditional termination	116
3.7.3. Defining a conditional cycle	117
3.7.4. Defining a conditional visit	117
3.7.5. Defining a conditional plate	117
3.8. Page Map	118
3.9. Lookup Table and Missing Value Maps	119
3.10. QC and CRF Sort Map	121
3.10.1. Editing the QC and CRF Sort Map	121
3.11. QC Cover, Message, and Title Text	122
3.12. Exiting DFconfig	123
3.12.1. Exit status	124
Chapter 4. Status Tool	125
4.1. Introduction	125
4.2. Server Status	127
4.3. User Status	128
4.4. Database Status	128
4.5. File Menu	129

4.6. Actions Menu	129
Chapter 5. Validation Tool	131
5.1. Introduction	131
5.1.1. Validation of New Records	131
5.1.2. Validation of Old Records	131
5.1.3. Validation of Old Records Following a Setup Change	132
5.1.4. Quality Control Notes	132
5.1.5. Reasons for Data (Change)	133
5.1.6. Duplicate Faxes	134
5.1.7. Intelligent Character Recognition (ICR)	134
5.1.8. Validation Levels	134
5.1.9. Record Status Levels	135
5.1.10. Retrieval Sets and Record Locking	135
5.1.11. Retrieval Modes	136
5.1.12. Signing Off Data Records	136
5.1.13. Color Coding	136
5.2. Retrieving Data Records	137
5.2.1. Building a Retrieval Specification	139
5.2.2. New Records	139
5.2.3. By Data Fields	143
5.2.4. By DataFax Retrieval File	146
5.2.5. By Program	148
5.2.6. By Variables	148
5.2.7. By QC Report	151
5.2.8. By Double Data Entry Batch File	152
5.2.9. Raw Data Entry	152
5.2.10. Retrieval Tasks	157
5.3. Validating Data Records	159
5.3.1. Validation Tool Windows	159
5.3.2. Moving Through the Records in a Retrieval Set	165
5.3.3. Editing Data Fields	166
5.3.4. Signing Off Records: Setting Status and Validation Level	171
5.3.5. Resolving Duplicates During Validation of New Records	172
5.3.6. Resolving Duplicates During Validation of Old Records	179
5.3.7. Validating Returned QC Reports	179
5.3.8. Validation Using Double Data Entry	179
5.4. Validation Tool Menus	181
5.4.1. File Menu	181
5.4.2. Set Menu	184
5.4.3. Record Menu	191
5.4.4. QC Menu	195
5.4.5. Reason Menu	200
5.4.6. Options Menu	202
5.4.7. Misc Menu	202
5.5. Recommended Procedures	205
5.5.1. Validation Levels	205
5.5.2. Retrieval Sets	205
5.5.3. Validation Tool Status Window	205
5.5.4. QC Notes	205
5.5.5. Resolving Duplicates	206
5.5.6. Fax Noise	206
5.5.7. Color	206
Chapter 6. CRF Viewer	207
6.1. Introduction	207
6.2. Retrieving a Set of CRF Pages	207
6.2.1. Retrieval by Data Fields	208

6.2.2. Retrieval by Program	211
6.2.3. Retrieval by DataFax Retrieval File	211
6.2.4. Retrieval by Variable	212
6.3. CRF List Window	214
6.4. CRF Context Window	215
6.5. The QCs Window	215
6.6. Retrieval Sets	216
6.7. Record Locking	216
6.8. Printing CRFs	217
6.9. Comparing CRFs	217
6.10. Creating PDF Files	218
Chapter 7. Windows-based CRF viewing with DFlite	220
7.1. Introduction	220
7.2. DFlite Basics	220
7.3. Getting started with DFlite	224
7.3.1. Pre-requisites	224
7.3.2. Starting DFlite	224
7.3.3. Logging in	225
7.3.4. Selecting a Study	226
7.4. Using DFlite	226
7.4.1. Tool Windows	227
7.4.2. CRF Operations	229
7.5. Limitations	231
7.6. Troubleshooting	232
Chapter 8. QC Tool	233
8.1. Introduction	233
8.2. QC Notes	233
8.3. The QC Database	233
8.4. Reviewing QC Status	233
8.4.1. Selecting a Set Of QC Notes	234
8.4.2. Update - Periodically	236
8.4.3. Update - Reset	236
8.4.4. Options - Exact Update and Quick Update	236
8.4.5. Options - View QC Notes	237
8.4.6. File - Print Status and Print Notes	238
8.4.7. File - Save Status As and Save Notes As	239
8.4.8. File - Save DataFax Retrieval File	240
Chapter 9. Lost Data Log Tool	241
9.1. Introduction	241
9.2. Creating Lost Data Records	241
9.3. Adding Records To DFidl	243
9.4. Retrieving Lost Data Records	243
9.5. Deleting Lost Data Records	244
9.6. Printing and Saving Lost Data Log Records to a File	244
9.7. Duplicate Resolution	244
Chapter 10. Reports Tool	245
10.1. Introduction	245
10.2. A Report Example	245
10.3. Report Types	246
10.3.1. DataFax Generic Reports	247
10.3.2. Study Specific Reports	247
10.3.3. History List	247
10.4. Standard DataFax Reports: Synopsis	247
10.4.1. Audit Trail	248
10.4.2. Center Tracking	248
10.4.3. Integrity Checks	248

10.4.4. Miscellaneous	248
10.4.5. Patient Tracking	248
10.4.6. Quality Control	248
10.4.7. Study Setup	249
10.4.8. Work Flow	249
10.5. Using DFreports	249
10.5.1. Selecting a Report	249
10.5.2. Setting Report Options	250
10.5.3. Running a Report	252
10.5.4. The Scrolling Output Window	252
10.5.5. Saving and Printing Report Output	252
10.5.6. Resetting the Reports Tool	252
10.5.7. Running reports from the command line	252
10.5.8. Reading On-line Documentation for Reports	253
10.5.9. Creating On-line Documentation for Study Specific Reports	254
10.6. DataFax Quality Control Reports	255
10.6.1. Introduction	255
10.6.2. Before Creating New Quality Control Reports	258
10.6.3. Types of Quality Control Reports	262
10.6.4. Quality Control Report Creation Options	264
10.6.5. Executing Quality Control Reports	269
10.6.6. Checking Quality Control Reports	271
10.6.7. Faxing Quality Control Reports	272
10.6.8. Checking Fax Transmission Status	273
10.6.9. Registering Successfully Faxed QC Reports	273
10.7. Installing Your Own Reports	276
Chapter 11. Data Export Tool	279
11.1. Introduction	279
11.2. A Quick Start Guide	279
11.2.1. Starting DFexport	279
11.2.2. User permissions	279
11.2.3. User interface	279
11.2.4. Minimum specifications	281
11.2.5. To narrow your search	281
11.2.6. Saving a retrieval	281
11.2.7. Output options	282
11.2.8. Searching the output	282
11.2.9. Error handling	282
11.3. DFexport Tasks	282
11.3.1. Create an Export Task	282
11.3.2. Modify an Export Task	284
11.3.3. Delete an Export Task	284
11.4. Plate Selection	284
11.5. Field Selection	285
11.5.1. Exporting Key Fields	285
11.5.2. Exporting by Field Number	286
11.5.3. Exporting by Field Name	287
11.5.4. The Field Editor Dialog	287
11.6. Data Filters	289
11.6.1. Status (DFexport.rpc option -s)	289
11.6.2. Validation (DFexport.rpc option -v)	291
11.6.3. Center (DFexport.rpc option -n)	292
11.6.4. Patient (DFexport.rpc option -I)	292
11.6.5. Visit (DFexport.rpc option -V)	292
11.6.6. Creation date (DFexport.rpc option -C)	292
11.6.7. Modification date (DFexport.rpc option -M)	292

11.6.8. Query	292
11.6.9. Pre-Defined Filters	294
11.7. Date Formats	296
11.8. Options	297
11.8.1. Decode Coded Fields	297
11.8.2. Quiet Mode	299
11.9. Lost Data Codes	300
11.10. Data Output Options	300
11.10.1. Screen	300
11.10.2. Write to File	300
11.10.3. Append to File	300
11.10.4. Print Header	300
11.10.5. Comma Separator	301
11.10.6. Trailing Pipe	301
11.11. Output Window	301
11.12. Actions	302
11.12.1. Run	302
11.12.2. Clear Output	302
11.12.3. Search and Find	302
11.12.4. Sort	302
11.13. Title Bar Menus	302
11.13.1. File	302
11.13.2. Edit	303
11.13.3. Search	303
11.13.4. Sort	303
11.13.5. Filter	304
11.13.6. Misc	304
11.13.7. Help	304
Chapter 12. Unidentified Fax Router	305
12.1. Introduction	305
12.2. DFrouter Functions	305
12.3. Starting DFrouter	306
12.4. The File Menu	307
12.4.1. Update Record List	307
12.4.2. Exit	307
12.5. The Page Menu	308
12.5.1. Rotate	308
12.5.2. Shift	309
12.5.3. Flip	310
12.5.4. Truncate	310
12.5.5. Trim Width	311
12.5.6. Rotate 90° CW/CCW	312
12.5.7. Rotate 90° CW/CCW and scale	312
12.5.8. Cut	313
12.5.9. Previous	313
12.5.10. Next	313
12.5.11. Print	313
12.5.12. Context	314
12.5.13. Reset	315
12.6. The Action Menu	315
12.6.1. Rotate/Shift/Identify CRF	315
12.6.2. Identify CRF	316
12.6.3. Notify User	316
12.6.4. Move	317
12.6.5. Discard	318
Chapter 13. Personal Faxmail Viewer	319

13.1. Introduction	319
13.2. DFfaxview Functions	319
13.3. DFfaxview Main Window	319
13.4. The File Menu	320
13.4.1. Load In Box	320
13.4.2. Print	320
13.4.3. Exit	321
13.5. The Action Menu	321
13.5.1. Discard	321
13.5.2. Send To Router	321
Chapter 14. Customizing appearance and behavior	322
14.1. Introduction	322
14.2. Background	322
14.2.1. Naming Resources	323
14.2.2. Locating resources	323
14.3. DataFax defaults	324
14.3.1. Application-level defaults	324
14.3.2. Study-level defaults	324
14.3.3. User-level defaults	325
14.3.4. Defaults by application	325

DataFax User Guide

Release 3.8.2

Copyright © 2007 Clinical DataFax Systems Inc

All rights reserved. No part of this publication may be re-transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Clinical DataFax Systems Inc. Permission is granted for internal re-distribution of this publication by the license holder and his employees for internal use only, provided that the copyright notices and this permission notice appear in all copies.

The information in this document is furnished for informational use only and is subject to change without notice. Clinical DataFax Systems Inc assumes no responsibility or liability for any errors or inaccuracies in this document or for any omissions from it.

All products or services mentioned in this document are covered by the trademarks, service marks, or product names as designated by the companies who market those products.

30 May 2007

Abstract

This guide has been written for those members of the data management team whose responsibility it is to use DataFax to manage one or more clinical trials. This guide describes how to setup DataFax for a new study and how to use it in the day-to-day management of the trial.

Preface

1. Getting Help

Clinical DataFax Systems Inc's Technical Support Department can be contacted Monday to Friday between 9:00 am and 5:00 pm Eastern Standard Time via any of the following methods:

25 Main Street West, Suite 500
Hamilton, Ontario L8P 1H1
Telephone: (905)-522-3282
Fax: (905)-522-7284
Email: <support@datafax.com>
URL: www.datafax.com (<http://www.datafax.com/>)

2. DataFax Mailing List

Clinical DataFax Systems Inc provides an automated email mailing list for tips, help, and interaction with other DataFax users.

To subscribe to the mailing list, complete the simple form found at the DataFax User's Group mailing list (<http://www.datafax.com/mailman/listinfo/dfug>) webpage. It is also possible to unsubscribe from the mailing list by visiting the same webpage.

To submit a message to the mailing list

1. Subscribe to the list. Message submissions are only accepted from members of the list.
2. Create your email message.
3. Send the email message to <DFUG@datafax.com>.

3. Conventions

A number of conventions have been used throughout this document.

Any freestanding sections of code are generally shown like this:

```
# this is example code  
code = code + overhead;
```

Text may also have several styles:

- Emphasized words are shown as follows: *emphasized words*.
- Filenames appear in the text like so: `dummy.c`.
- Code, constants, and literals in the text appear like so: `main`.
- Variable names appear in the text like so: `nBytes`.
- Text on user interface labels or menus is shown as: Printer name, while buttons in user interfaces are shown as Cancel.

If the current document identifies changes from an earlier version ^[1], they are identified as:

- this is example text for changes
- this is example text for additions
- this is example text for deletions

^[1] If this document does not identify changes, all of the examples will have the same appearance.

Chapter 1. Introduction

1.1. Why DataFax?

It might help you to understand DataFax a little better if we start with a brief explanation of our major objectives. DataFax represents our attempt to solve the logistical problems associated with managing large multicenter clinical trials. Its principal goal is to improve the 2-way flow of information between the clinical sites, where patients are seen, assessed and treated, and the coordinating center, which is responsible for running the trial according to an approved protocol, collecting patient data from the clinical sites, correcting errors and omissions as required, and in the end, producing a final analysis and interpretation of the study results.

DataFax starts from the assumption that paper case report forms (CRFs) will continue to be the easiest and most cost effective way for busy health care professionals, working in a variety of health care and community settings, to record patient data. Starting from this assumption, our primary objective in creating DataFax was to automate the collection and processing of paper case report forms, with the ultimate goal of improving the timeliness and quality of the study database. Our specific design objectives were to:

- **Use simple technology in the clinical sites.** Clinical sites send CRFs to the data management center using ordinary fax machines.
- **Computerize the receipt, logging and filing of CRFs.** In the data management office the faxed CRFs are received by a fax modem connected to a computer with software which can recognize which study the CRFs belong to, queue them up for data entry and store them to optical or magnetic disc.
- **Automatically generate an initial data record.** DataFax reads data boxes (Xs, handwritten numbers and VAS) to create an initial data record as the starting point for the clinical review and data validation process.
- **Provide split screen review of CRFs and the corresponding data records.** All faxed CRFs and initial data records are reviewed on screen to complete data entry (text fields) and to make corrections and flag problems (e.g. missing data).
- **Automate the quality control process.** Problems detected on the faxed CRFs are flagged using quality control notes (electronic post-its) which are automatically formatted into standard quality control reports for transmission by fax to the clinical sites.
- **Automate work flow management.** CRFs are stamped with a validation level at each CRF review and data processing stage. The current status of all faxed CRF pages is reported by the Status Tool (**DFstatus**) and data records at a particular validation level can be retrieved for the next level of review.

1.2. About This Guide

This guide has been written for those members of the data management team whose responsibility it is to use DataFax to manage one or more clinical trials. This guide describes how to setup DataFax for a new study and how to use it in the day-to-day management of the trial.

It does not describe DataFax system administration or the things you need to consider when planning to run a new study using DataFax. These topics are covered in the *DataFax System Administrator Guide* and the *DataFax Study Planning Guide* respectively. It assumes that the study CRFs have been created in DataFax format, and that your DataFax administrator has registered the new study and the staff who will be working on it.

We will start with a brief introduction to some basic DataFax concepts and then describe the user interface design. Subsequent chapters will describe each tool in the DataFax Tool Box. The order of presentation is pretty much the order in which you will actually use the tools during a normal working day.

The DataFax tools are built to run on top of the X11R5 and Motif libraries, but they can be displayed in any X11 compliant window manager. All of our examples use the Motif Window Manager (MWM) or the Common Desktop Environment (CDE) but the examples will work equally well with other X window manager.

We have tried to minimize the use of specialized terms but some DataFax jargon is regrettably unavoidable. The following section, Basic Concepts, introduces most of the essential ideas and terms.

1.3. Basic Concepts

1.3.1. DataFax CRFs

DataFax does not work with arbitrary case report forms (CRFs). Study CRFs must be designed to DataFax specifications which include:

- **Bar coding at the top of each CRF page.** To identify the study, the CRF plate, and optionally the sequence (or visit) number.
- **Page format.** All pages must be US letter or A4 size and oriented vertically (portrait) not horizontally (landscape). [2]
- **Numerical data entry fields.** All boxes designed for numerical data must be square, a quarter inch on each side, and the first box in any sequence of adjacent boxes, must be preceded by at least one-eighth inch of white space. The automated recognition of numbers only works if numbers are kept entirely inside the boxes on the data forms.

1.3.2. DataFax Plates

The entire case record book (CRB) for a study typically consists of a number of unique pages, some of which are repeated over time. Each unique page of the CRB (defined by contents and format) corresponds to a DataFax plate. For example, the plate on which medications or adverse reactions are recorded will have the same contents and appearance across all follow-up exams, i.e. the same plate is used each time. The specific instance of each plate is identified by a sequence number which usually corresponds to a scheduled patient visit number.

It is possible to give every page of the CRB a unique plate number (provided the total number of pages is 500 or less for each study). This will create a different data file for each plate/visit combination. However, it is more efficient and convenient to only assign unique plate numbers to pages which differ in contents and/or format.

1.3.3. DataFax Database

The database maintained by DataFax for each study is made up of simple ASCII files, with one file corresponding to each plate in the study CRB. Within data files, each record corresponds to one copy of the plate, which is uniquely identified by the patient ID number and the sequence (or visit) number. Thus, each data file holds the data recorded on all instances of the plate (i.e. across all patients and all follow-up visits). This is sometimes referred to as a page-based database design because there is one data record for every physical CRF page received during the study.

1.3.4. DataFax Keys

Each record in a DataFax data file has 4 keys which identify the particular study, patient, plate, and sequence (or visit) number that the data belongs to. These keys must appear in the following order at the top of every CRF page.

1.3.4.1. Key #1: Study Number

The study number is the first key field on each CRF page. It has a value for each study database, in the range 1 to 255, and must appear in the barcode at the top of the page. Study numbers can also be thought of as unique database identifiers. For example, within one project you might have the main patient database, consisting of all data gathered on patient CRFs, and a second database consisting of information related to the clinics participating in the trial. In addition to using different plates to define the data collected in these 2 databases, they will also be distinguished by the use of a different case ID number (i.e. patient IDs vs center IDs).

1.3.4.2. Key #2: Plate Number

The plate number is the second key field on each CRF page. Each plate within a study database has a unique value, in the range 1 to 500, and must appear in the barcode immediately after the study number. Remember a plate is a template or layout that is unique in content and presentation. There might be several pages with the same plate number but a different visit/sequence number.

1.3.4.3. Key #3: Visit/Sequence Number

The visit or sequence number is the third key field on each page. When the same data is collected at different times using the same CRF plate, the visit or sequence number is used to distinguish the possible multiple occurrences of the same plate for the same patient.

The visit/sequence key can appear either as the third component of the bar code, immediately after the plate number, or as the first data field defined on the page. It can only be barcoded if the value can be fixed on the CRF page in advance (e.g. baseline: visit# 0). Visit/sequence numbers can be any integer in the range 0-65535, however only values in the range 0-511 can be barcoded.



Important

All keys are required. Thus, even if a particular plate occurs on only one page and there is no need to indicate repetitions over time, you still need to assign the plate a visit/sequence number.

1.3.4.4. Key #4: Patient ID Number

The Patient (or case) ID number is the fourth and final DataFax key field. It must appear immediately following the visit or sequence number. It can not appear in the barcode. It must be the first data field (on pages with the visit/sequence number in the barcode), or the second data field (on pages with the visit/sequence number in the first data field).

Patient ID numbers can be any integer in the range 1-2147483647. In multicenter clinical trials it is helpful to design the patient ID number in 2 parts, with the first 2 or 3 boxes for the center or clinic number, followed by another 2 or 3 boxes to identify patients within each center. It is helpful to separate these 2 parts with a bit of space on the study CRFs to improve readability, but they must be defined together as making up a single data field, when the Patient ID data field is defined in **DFsetup**.

1.3.5. Record Status

At the end of each step in the validation process, data records are signed off by setting their status field. This identifies whether there are any problems on the corresponding CRF page, and whether that page is the current (most up to date) copy of the page or an older version. The possible status categories are as follows:

- **pending.** Initial data entry of the record has commenced but has not been completed.
- **clean.** There are no problems with the record.
- **dirty.** One or more data fields are flagged with unresolved quality control notes.
- **error.** One or more keys (plate, sequence or patient) could not be identified.
- **CLEAN.** An older version (secondary record) with status set to clean.
- **DIRTY.** An older version (secondary record) with status set to dirty.
- **ERROR.** An older version (secondary record) with status set to error.
- **Lost.** CRF pages which are not expected (patient refused, died, etc.).

1.3.6. Duplicate Copies of a CRF Page

Duplicates arise when a CRF page is faxed into the study database more than once. Duplicates are recognized and resolved by marking one copy (the most recent and up to date) as the primary version. Other copies can either be retained as secondary versions or deleted. Duplicates may arise because of a refax request sent to the center on a quality control report, or because the investigator made a correction that was unsolicited and then faxed in the revised page. Usually the most recent version of a CRF page to arrive will be the most complete and up-to-date and thus will be accepted as the new 'primary' record, and the previous version will be saved as a 'secondary' record.

1.3.7. Validation Levels and Work Flow Management

DataFax helps to automate work flow management through the use of validation levels. Data records are stamped with a validation level at each step in their flow through the various reviews set up by the study data management team. When a new fax arrives each page is read by the ICR software and the resulting data records are given a validation level of 0. DataFax then allows up to 7 validation levels which each record can go through as various members of the data management team perform their reviews of the CRFs.

When a new study is set up each user who is authorized to access the study database is assigned a maximum validation level for which they can enter data. However, it is possible for a user with a lower validation level to retrieve and re-enter a record that exists at a higher validation level, the result being that the record is assigned the lower validation and re-enters the workflow at that lower level. For example, when a duplicate of an existing page of the CRF arrives in the database, the first data clerk needs to be able to accept it. When this occurs the validation level for that page is reset to the first clerk's validation level (usually 1) and the record will then proceed again up through the various validation levels defined for the study.

Users need not always work at their maximum validation level. They may work at lower validation levels if the specific task they are performing requires it. As an example suppose that a study has 2 data entry clerks whose primary role is to validate all new fax arrivals and to double check each others work. One way to organize these two individuals would be to give both of them a maximum validation level of 3. When validating newly arrived faxes one of them could be instructed to work at validation level 1 and the other at validation level 2. For the double validation step both clerks would work at validation level 3 but the first data clerk would retrieve all records which were at validation level 2 and the second clerk would retrieve all records which were at validation level 1. This would guarantee that they would always double check each others work and never simply review their own.

Normally, higher level reviews are performed after lower level reviews have been completed. However, sometimes it may be necessary to skip lower level reviews and proceed directly with a higher level review. For example, suppose that a CRA with a maximum validation level of 4, has responsibility for reviewing sensitive and critical information. The CRA reviews all adverse event reports as soon as possible, (i.e. regardless of validation level), all entry lab reports once they had reached validation level 3, and all records with new QC notes before quality control reports are created and faxed to the investigators. For all of these tasks the CRA would work at validation level 4. When the review of each record is completed it is signed off by the CRA to promote it to validation level 4, thus signifying that his/her review has been completed.

Thus an adverse event report might skip from validation level 1 or 2 directly to level 4. But since immediate review is important and the CRA review is more critical than that of the second data entry clerk this is appropriate. By only reviewing the lab reports when they reach level 3 the CRA would have the advantage of the prior double validation by the data entry clerks and thus would expect to find fewer problems that needed his/her attention. When reviewing all records with new QC notes the CRA might again sometimes elevate a record from level 1 or 2 to level 4 (unless the volume of incoming traffic was low enough that the double check could always be completed before new quality control reports were created). The CRA would thus have to take care to review all of the page, not just the fields with QC notes attached, for the record to have undergone a proper double review.

In addition there might be a senior clinical project manager who reviews a random sample of records as a final quality control step. This person might work at level 5 and only review records which were at levels 3 and 4. As a final step there might be an operation which exports clean records which have reached their maximum review level (3,4 or 5 in this example) to some other data management system. The end result of this operation might be to raise these records to level 6 to signify that they had been exported.

Using this system you would know that records which were at levels 1 and 2 had only been through a single validation, records at level 3 had been double validated, records at levels 4 and 5 had been reviewed by the CRA and clinical project leader respectively, and records at level 6 had been exported to another database management system.

The Status Tool (**DFstatus**) provides a quick method of reviewing both the status and validation level of all records in the study database.

1.4. Starting DataFax

The DataFax user toolbox is the user interface to the DataFax system. There are two common ways to start the DataFax toolbox:

- **Start the toolbox from a command prompt.** The DataFax program is in the DataFax `bin` directory (which you do not need to type if the `bin` directory is in your path).

You will probably want to start the toolbox as a background process with the command:

```
% $DATAFAX_DIR/bin/DataFax &
```

If this command generates an error message or the DataFax toolbox appears but does not look right, then it is likely true that your login environment has not been initialized for DataFax. To initialize your DataFax environment and rectify this, type the following command (your DataFax software may not be in `/opt/datafax`, so modify this accordingly):

```
% /opt/datafax/utils/DFmkUserEnv
```

This will initialize your DataFax environment. For the changes to take effect, you will need to logout and log back in again. Then try starting DataFax again with the command:

```
% $DATAFAX_DIR/bin/DataFax
```

- **Launch the toolbox from the workspace of your window manager.** You may need to customize your window manager to do this. The exact steps to do this will vary dependent on the window manager that you are using.

1.5. X11/Motif and DataFax User Interface

DataFax is an X11R5/Motif compliant application which means that it can be displayed in any window environment that understands the X Window System protocol. We won't try to cover all of the X11 or Motif user interface in this document. There are many good books available that describe the complete user interface. Instead, we will describe the user interface elements that are common to all of the DataFax tools.

1.5.1. Using the Three Button Mouse

The Motif design is fairly flexible in the way that it allows mouse buttons to be bound to actions other than the default. However, since this task is not for the faint of heart, we will assume that all users will use the default mouse button bindings. Motif also assumes that your mouse has three mouse buttons. In Motif terminology, the mouse buttons are not actually named *Left*, *Middle*, and *Right*, but instead are named:

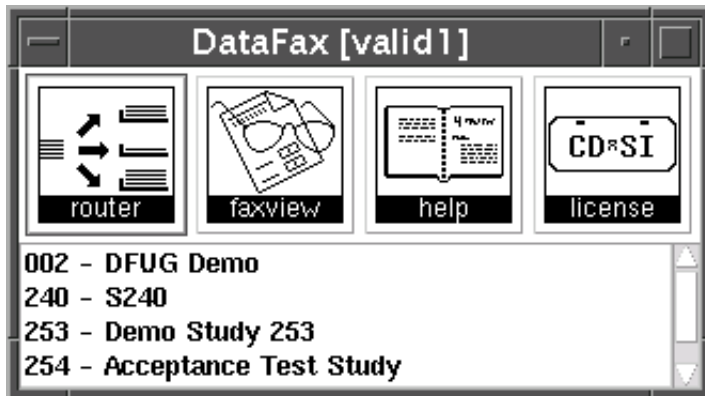
<code>BSelect</code>	Used for selection, activation, and setting the location cursor. This button must be the leftmost button, except for left-handed users where it can be the rightmost button.
<code>BTransfer</code>	Used for moving and copying elements. This button must be the middle mouse button.
<code>BMenu</code>	Used for popping up menus. This button must be the rightmost button, except for left-handed users, where it can be the leftmost button.

If you have a two-button mouse, `BSelect` must be the leftmost button, and `BTransfer` is bound to the rightmost button. `BMenu` is then bound to `AltBSelect`, which means hold down the Alt key while pressing the left mouse button. Some PC-based X windows servers also emulate the middle mouse button on a two-button mouse when both mouse buttons are depressed simultaneously.

The `BSelect`, `BTransfer`, and `BMenu` button binding model is conceptually very elegant but it is not necessarily intuitive for the reader who is used to seeing references to the *Left*, *Middle*, and *Right* buttons. For this reason we will stick with the *Left*, *Middle*, and *Right* names in our documentation. However, you must remember this and make the mental translation yourself if you use non-default button bindings.

1.5.2. DataFax User and Study Toolboxes

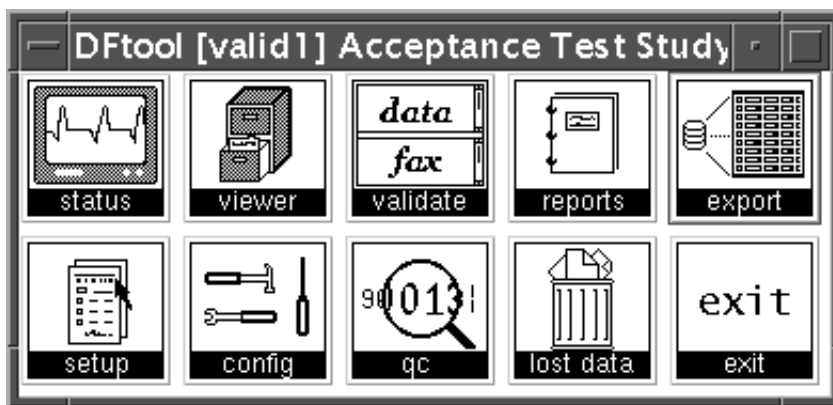
The DataFax User Toolbox is the first-level DataFax Toolbox from which you launch all of your other DataFax tools. The User Toolbox includes tools that are not specific to any study such as the Unidentified Fax Router (**DFrouter**) and the Fax Viewer (**DFfaxview**). It also includes a scrolling list of all of the studies that you have permissions in DataFax to access.

Figure 1.1. The DataFax User Toolbox

To start **DFrouter** or **DFfaxview** programs, simply click once on their respective icons. When an icon is selected from the DataFax Toolbox, the shadows on the icon are reversed so that the icon button no longer appears to be "sticking out". This indicates that the program has started. When the program exits, the original shadows on the icon are restored indicating that the program has stopped and can once again be selected.

To start a Toolbox for a particular study, locate the study entry in the scrolling list. If the study does not appear in the list then you do not have permissions to access that study. If this is an error, your DataFax system administrator can give you permissions to access the study. If the study does appear in the list, double-click on the study entry to start the Toolbox. It may take a few seconds for the Toolbox to appear so be patient. If you had previously launched a Study Toolbox and subsequently iconized the Toolbox, double-clicking on the study entry will maximize your previous Toolbox and not start another one.

The DataFax Study Toolbox has the following appearance.

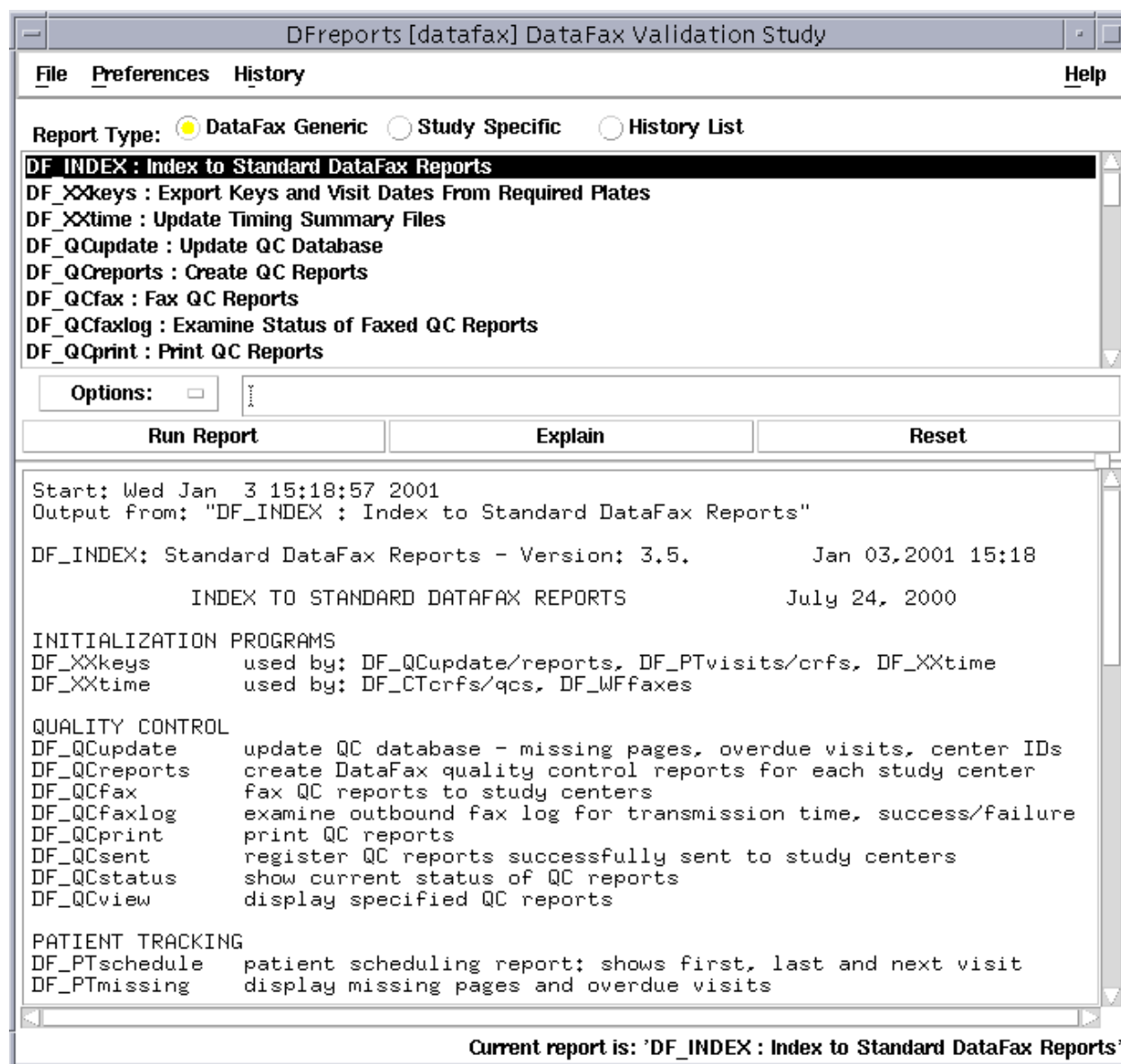
Figure 1.2. The DataFax Study Toolbox

All DataFax tools can be invoked by selecting with a single mouse click the appropriate icon from the Study Toolbox. When invoked, each tool brings up a different window, often with sub-windows as needed, in which the work of that tool is directed by you and displayed by DataFax. Note that once invoked, re-clicking on a tool icon results in the tool remaining open and causes the main tool window to be mapped and raised. If the main tool window has been minimized, re-selecting its icon will restore the main window from its minimized state.

1.5.3. Common Interface Components

Each DataFax tool window has several interface components in common with the other DataFax tool windows. This is no surprise as every window application has certain characteristics in common with other window applications, such as window decorations, menu areas, input areas, output areas, and message areas. For example, the Reports Tool (**DFreports**) has most of the common interface components that you can expect to see in any DataFax application.

Figure 1.3. Common DataFax Interface Components in DFreports



1.5.4. Mnemonics

Each of the functions accessible from the menu bar of a DataFax tool has a corresponding mnemonic which can be typed at the keyboard to activate the menu item, as though it was chosen with the mouse.

The steps to activating a mnemonic are as follows:

1. Press the F10 function key.

This moves the keyboard focus to the menu bar and highlights the leftmost menu item.

2. Enter the mnemonic for the menu that you wish to choose.

The mnemonic is the underlined letter (case in-sensitive) in the name of the menu. Type this letter. This highlights the menu item with that letter for a mnemonic and cascades (pops down) the menu. For example, to choose the Preferences menu in the **DFreports** window above, type 'p'.

3. Enter the mnemonic for the menu item that you wish to choose.

Again, the mnemonic is the underlined letter in the name of the menu item. If the item has no pullright menus, the action associated with the item is executed. If the item has a pullright menu, the pullright opens, and step 3. is repeated.

You can cancel the entry of a mnemonic by pressing the Esc key at any time.

1.5.5. Dialog Boxes

Menu items that have an ellipsis (...) at the end of their name will display a dialog box when they are chosen. A dialog box is a popup window that requests additional information before the action in the menu item can be completed. A dialog box may contain one or more text fields to be completed, choice fields from which an option may be selected, and/or other action buttons. The QC Add dialog from **DFvalidate** is a good example of a dialog box and is illustrated in Figure 1.4.

Figure 1.4. The QC Add dialog box in the Validation Tool.

DFvalidate – Add ...

Id: 1001 (Center: 1) Visit: 1 Plate: 2

Variable Medication Code #

Value 1143

Problem Other ☐ **Use** External ☐

Status New ☐ **Type** Refax ☐

Query ...

Reply

Note

Created

Modified

Resolved

OK **Cancel**

Interacting with a dialog box involves specifying values for the additional fields in the dialog, and then applying those specifications by selecting OK. You can cancel a dialog and the menu action that originated it by selecting Cancel.

1.5.5.1. OK, Apply, and Cancel Actions

All dialog boxes have one or more action buttons in an action area at the bottom of the dialog. The most common action buttons are OK, Apply, and Cancel. OK causes the settings and modifications that you have specified to be applied, and then causes the dialog to pop down (disappear). Apply has the same effect except that the dialog box is not popped down after the changes have been applied. This can be useful for dialogs that are used frequently. Cancel causes the dialog to pop down without any changes being applied.

1.5.5.2. Common Dialog Boxes

There are a few dialog boxes that are common to each DataFax tool. These include message or warning dialogs, question dialogs, and the Print dialog.

1.5.5.2.1. Message and Warning Dialogs

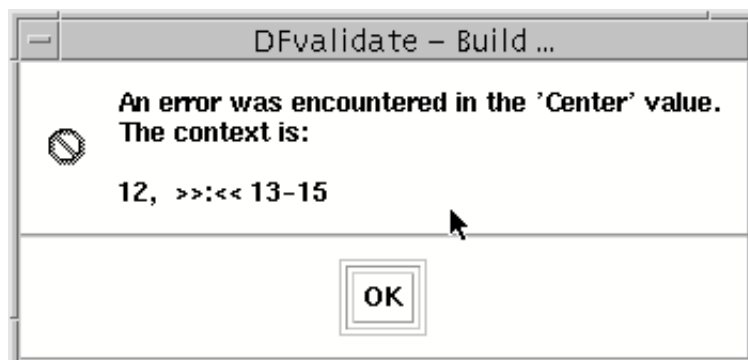
A message dialog conveys a message to the user, usually that some action has completed. A message dialog is not modal; that is, it does not interrupt your interaction with the application. A message dialog usually contains an information symbol, a message, and a single button, usually an OK button.

Figure 1.5. A DataFax Message Dialog Box



A warning dialog alerts the user to a possible danger. It is application modal; that is, it interrupts your interaction with the application but not with other applications. A warning dialog contains a warning symbol, a message, and one or two buttons, typically OK and Cancel.

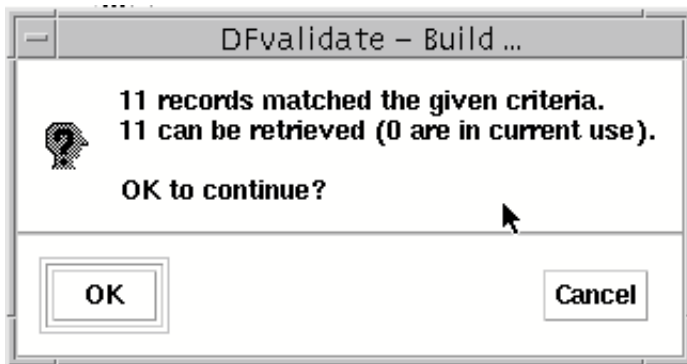
Figure 1.6. A DataFax Warning Dialog Box



1.5.5.2.2. Question Dialog

A question dialog is used to get a user response to a question. It is application modal, and includes a question symbol, a message or question, and two response buttons.

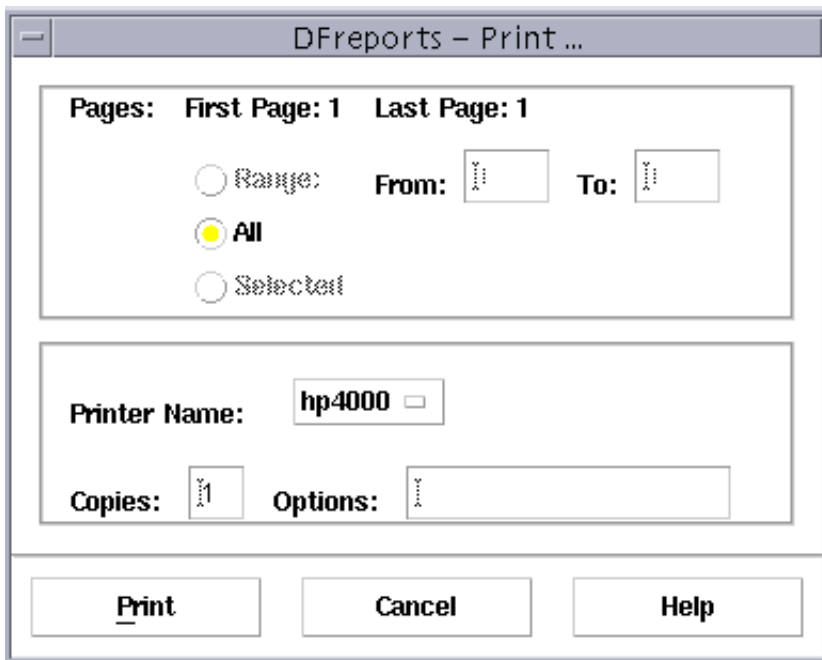
Figure 1.7. A DataFax Question Dialog Box



1.5.5.2.3. Print Dialog

The DataFax Print dialog allows you to control which pages are to be printed, how many copies of each page are to be printed, and which printer is to be used for the printing.

Figure 1.8. The DataFax Print Dialog Box



You can specify the pages to be printed as a range of pages, all of the pages, or as the current selected set of pages, typically selected from an external scrolling list of pages. If any of the options is inappropriate for the application, it is made insensitive. The Options field can be used when printing wide text files. Options should be specified as -p# where -p# represents the point size of the printed text. For example, -p8 specifies that the printed text will be in point size 8.

1.5.5.3. Text Fields

Text fields are used when entry and or display of a text string is required. A text field consists of an optional label and a value. Not all text fields allow entry of a value, some are display only. Some text fields allow entry of a text string longer than the width of the display field. Normally, when typing into a text field, the insertion cursor starts at the left and moves to the right when characters are entered. When the display field is full, typing further characters will cause the text to the left of the insertion cursor to scroll left. This will cause the left most portion of the string to become hidden.

1.5.5.3.1. Text Field Cursor

Text fields have a | cursor at the insertion position. Anything you type in a text field will be inserted immediately to the left of this cursor. The Delete key can be used to delete characters to the right of the cursor and the Backspace key to delete characters to the left of the cursor. The text insertion cursor can be moved with the mouse by selecting a new position in the text field, or it can be moved with the right and left arrow keys.

1.5.5.3.2. Copy and Paste in Text Fields

X11/ Motif defines an entire selection mechanism that supports copy and paste between text widgets. Motif 1.2 also supports a drag-and-drop protocol between many types of widgets, not just text widgets.

For DataFax purposes, there is a simple X Window System copy and paste mechanism that is effective for most situations. To copy text, select the beginning of the source text with the left mouse button and then, while holding down the left mouse button, drag the mouse to the right highlighting the text to be copied. Release the left mouse button when the rightmost character in the source has been highlighted. The highlighted text becomes your selection for copying. To paste text, place the mouse cursor at the location of the paste destination and press the middle (**BTransfer**) mouse button. The selection is inserted into the destination immediately to the left of the cursor.

1.5.5.4. Grayed-out Fields and Menu Items

If a text field or menu item should not be accessed at a given point it will be made insensitive and grayed-out. Typically, a widget is made insensitive because in the current context it would be an error to activate it.

1.5.5.5. Exiting from DataFax Tools

To exit from a DataFax, choose Exit from the File menu. The File menu is always the leftmost menu in the menubar of every DataFax application.

Although not necessary, we recommend that you always quit out of DataFax tools as soon as you are finished with them. This ensures that computer resources and any study resources, such as locked records, are freed up as soon as possible.

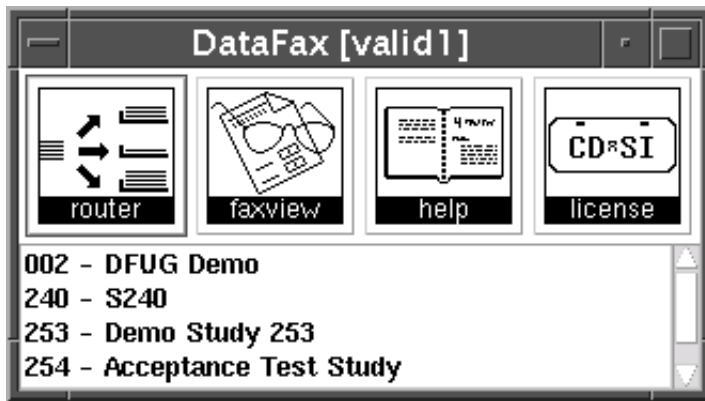
1.6. Obtaining Help with DataFax

1.6.1. Introduction

If you are working in a DataFax study and require information or assistance with any of the DataFax tools, you can select the Help icon from the DataFax User Tool Box (third icon from the left in the figure below). Selecting the help icon opens the DataFax documentation set in HTML format. Similarly, documentation for each of the DataFax tools may be viewed by first opening the tool from the Study Tool Box, then

selecting Contents from the Help menu located in the upper right corner of the tool window.

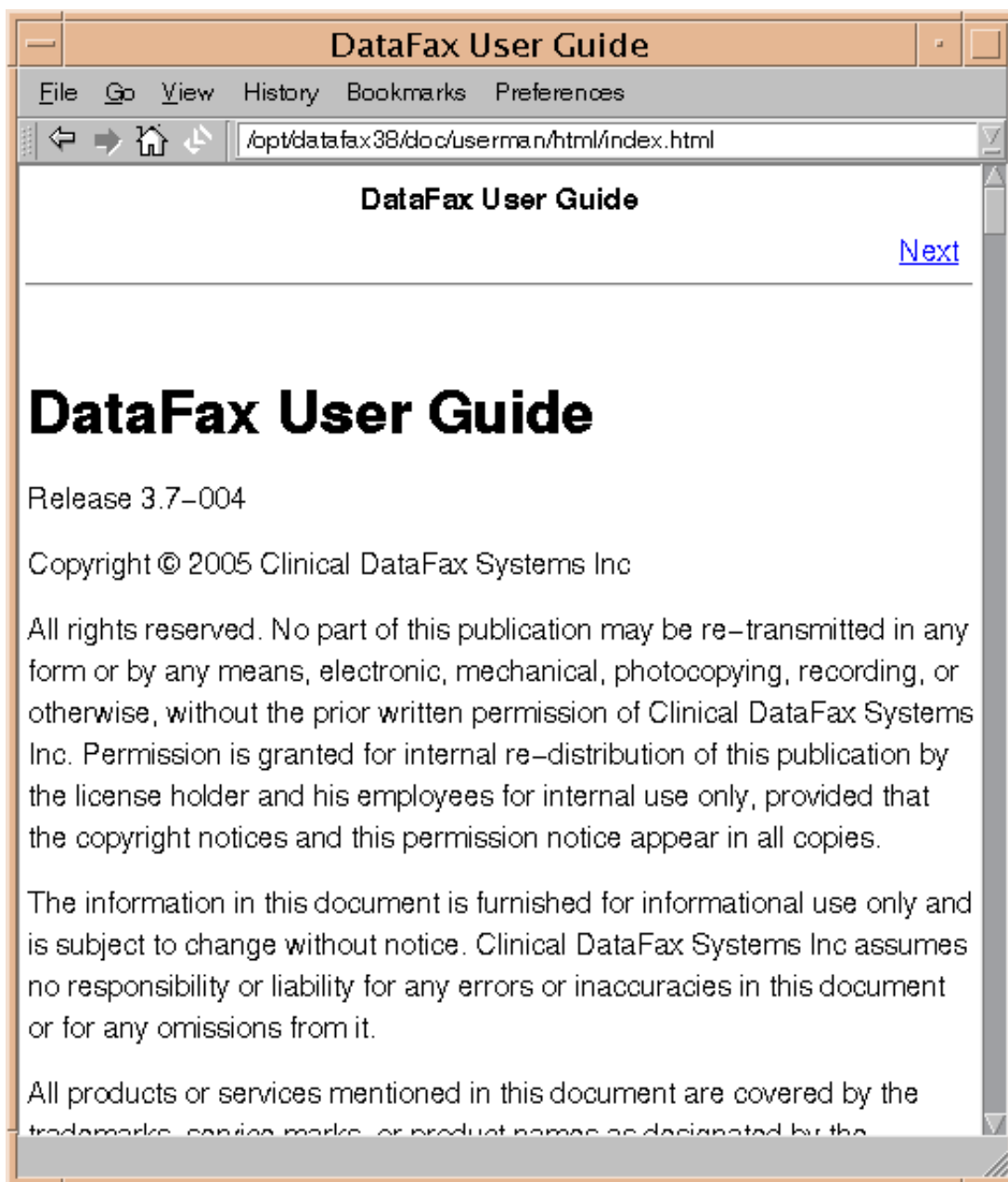
Figure 1.9. help can be started from the DataFax User Tool Box.



1.6.2. Using Help from the DataFax User Tool Box

Selecting help will start a browser-like window containing the DataFax documentation set.

Figure 1.10. The DataFax documentation set window as it appears once help is invoked



The following guides are available as on-line help:

- DataFax Study Planning Guide
- DataFax User Guide
- iDataFax User Guide
- DataFax Standard Reports Guide
- DataFax Programmer Guide
- DataFax System Administrator Guide
- DataFax Quick Reference Guide
- DataFax Software Installation Guide

Each of the above guides appear in 2 HTML formats: a single HTML file and multiple HTML files, split at section boundaries. Files in this latter format contain the content of a single section of documentation with navigation to the previous and next sections. Clicking on the appropriate link will take you to the corresponding document. The single HTML version is conducive to printing and casual browsing, but has the drawback of taking longer to download and display, while the sectioned files load quicker and are better for printing individual topics.

Once you are linked to the DataFax guide you want, a table of contents follows the abstract. Clicking on any table of contents entry changes the context to that section.

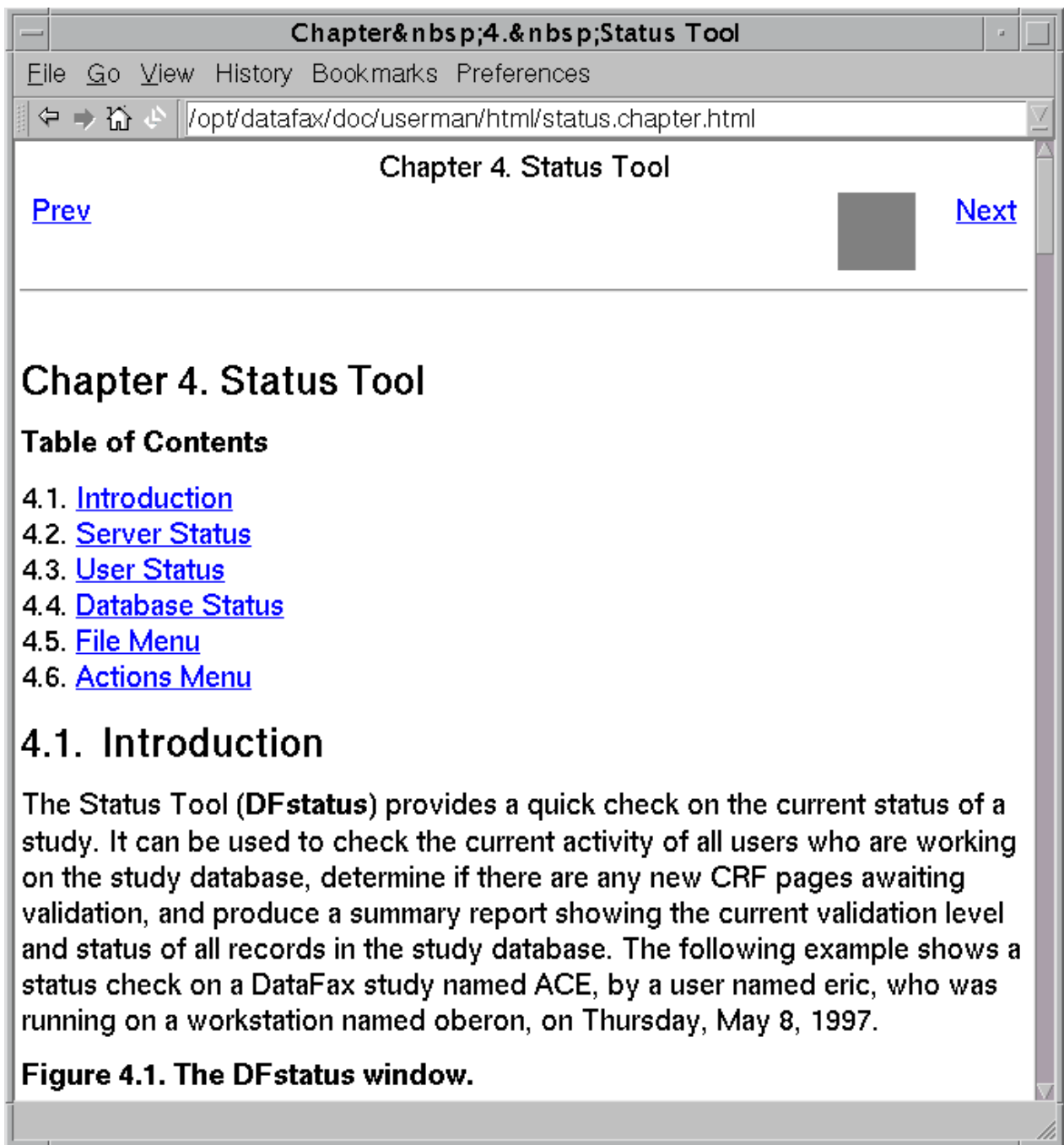
The DataFax documentation set available through help is the same as that found on the DataFax website www.datafax.com.

To exit Help, select Exit from the File menu at the top of the main Help window.

1.6.3. Using Help from the DataFax Study Tool Box

To obtain user documentation about a specific DataFax tool, open the tool by selecting the appropriate icon from the Study Tool Box. From the Help menu located in the upper right corner of the tool's main window, select > Contents. A single HTML file containing information about the selected tool will appear. Clicking on the appropriate link will take you to the corresponding section of the document.

Figure 1.11. The DataFax documentation for the Status tool appears once Help- > Contents is invoked



The DataFax documentation available through Help- > Contents is the same as that found in the DataFax User Guide on the DataFax website www.datafax.com.

1.6.4. Reading DataFax documentation with a web browser

The help viewer included with DataFax is able to display HTML content but does not have many of the features of mainstream web browsers such as Microsoft's Internet Explorer and Netscape's Navigator.

It is possible to read the DataFax documentation with a web browser by pointing the web browser to the root HTML file for the documentation. This root file is
\$DATAFAX_DIR/doc/allman/html/index.html.

[2] Beginning with release 3.8, it is possible to manually rotate a landscape page to portrait orientation and shrink it so that it fits in the width of a portrait page.

Chapter 2. Setup Tool

2.1. Introduction

The Setup Tool (**DFsetup**) is used to define each new DataFax study. Starting with a PostScript® or PDF file containing your study CRFs, **DFsetup** is used to create all of the study definition files DataFax requires to run a new study. These include the following:

- **Database schema.** The data dictionary which defines all data fields.
- **ICR template.** Identifies the location of data fields on each CRF plate.
- **Setup save file.** All setup specifications including style and variable definitions.

Creation of the database schema, ICR template and data entry screens are all accomplished simultaneously by simple point and click operations. You start by importing one copy of each unique plate making up the study CRFs, into **DFsetup**. Then for each plate, all data fields are located by clicking on them, and defined by specifying a variable name, description, coding for choice fields, formats for dates, legal values for numerical fields, help messages, etc. in a variable definition dialog. The ICR template is created automatically by **DFsetup** after you have defined the study database schema. The setup save file (created automatically by **DFsetup**) holds all plate, style and variable definitions. It is then updated by **DFsetup** whenever changes to the definition are made, and is read by the Validation Tool (**DFvalidate**) to define the data validation screens.

The DataFax Configuration tool (**DFconfig**) is used to create the study configuration files DataFax requires to manage the study sites and patient scheduling. These include the following:

- **Centers database.** Identifies all clinical sites participating in the study.
- **Visit map and conditional maps.** Defines patient assessment scheduling and CRFs to be completed.
- **Page map.** Provides CRF form names for use on Quality Control reports.

DFconfig is a simple editor tool that can be opened using the `config` icon in the study toolbox or by selecting Configuration for the Study menu in **DFsetup**. A detailed description of **DFconfig** can be found in **DFconfig**.

Anyone provided with a complete set of DataFax Study Setup Worksheets, which includes an annotated CRF (with variable names, coding, legal values, etc.), a list of the participating clinical sites, a clinic visit schedule, and a list of CRFs to be completed at each visit, should be able to define these files using **DFsetup** and **DFconfig**. No programming is required, not even to setup the data entry screens.

2.2. DataFax Database

A detailed description of the database structure maintained by DataFax can be found in the *DataFax Programmer Guide, The study data directory*. For now we simply note that DataFax generates a data record for each CRF page that arrives by fax and stores it in a data file corresponding to the CRF plate number barcoded on the faxed page. Further, the order of the data fields in each data record exactly matches the order in which they are defined on the corresponding CRF plate in **DFsetup**. Thus, it is worth remembering that when you design your study CRFs and when you use **DFsetup** to define variables, you are at the same time defining the structure of the DataFax database for your study.

2.3. Getting Ready

Before using **DFsetup** you must have completed the following steps:

1. Register the study.

Each study must be registered through the DataFax Administration Tool (**DFsystem**) before anyone will be able to use **DFsetup** to define the study definition files. Ask your DataFax system administrator to register the new study and grant permissions to study staff. Use the study configuration and user permissions form provided in the DataFax Study Setup Worksheets to provide your DataFax system administrator with the information that will be needed. Users who are granted permission to use **DFsetup** will automatically be granted permission to also use the configuration file editor **DFconfig**. Users who are not granted permission to use **DFsetup** will be granted view only access to **DFsetup** and **DFconfig** if they have been granted access to **DFreports**, otherwise they will not be able to access study setup specifications using these tools.

2. Finalize the study CRFs.

Although it is possible to make changes to the study setup at any time, even after the study has started, it is best to wait until the CRFs are finalized before using **DFsetup** to define your study. Remember to make the appropriate modifications to your setup for any CRF plates that are revised after setup is complete.

3. Save the study CRFs to a PostScript or PDF file.

Most desktop publishing packages can produce a PostScript or PDF output file. Although there are other ways of getting your CRFs into **DFsetup**, importing them from a PostScript or PDF file is the easiest method. Both these methods require the GhostScript software to be properly set up and for the user to have write permissions on the study work and bkgd directories.

4. Define Variable Styles for your study.

The use of variable styles simplifies variable definitions. See Variable Styles.

5. Prepare an annotated copy of the study CRFs.

It is a good idea to prepare an annotated copy of the study CRFs with variable names, styles, descriptions, coding, legal values and any special features like variable skip patterns and edit checks, before you start using **DFsetup**.

6. Define the study visit map, page map and other necessary configuration files.

A study visit map (**DFvisit_map**), and optionally a study page map (**DFpage_map**), should also be defined before you enter them using **DFconfig** in **DFsetup**. See *DataFax Study Planning Guide*, *Patient Visit Scheduling* and *DataFax Study Planning Guide, CRF Page Map*.

7. Assemble the information required on clinical sites.

To define the centers database (**DFcenters**) you will need a list of all of the clinical sites participating in the trial with the relevant information on each site as described in *DataFax Study Planning Guide*, *Clinical Centers Database*.

All of the above steps may be performed with the aid of the DataFax Study Setup Worksheets. Completing the worksheets ensure that all aspects of study setup and testing have been documented.

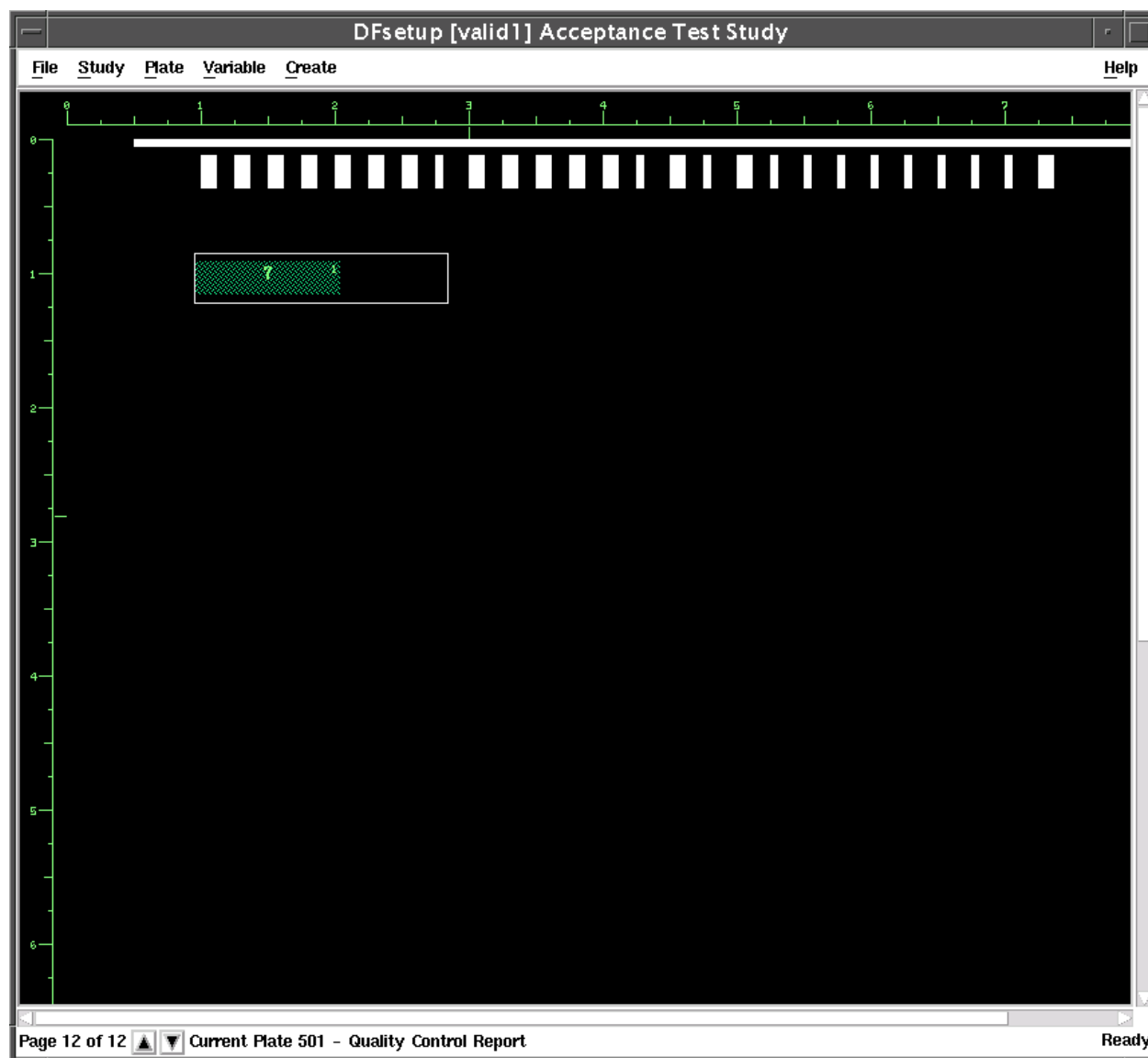
**Warning**

Although you can use **DFsetup** to add, delete or rearrange the data fields on CRF plates at any time, DataFax does not currently reformat any data records which might already exist in the database to match your modifications. A programmer must reformat existing data records for any modified plates. Thus before modifying the setup for a study which is in progress, carefully read and follow the instructions in Reordering Variables.

2.4. Starting a New Study Setup

Start **DFsetup** by clicking the setup icon in your study's DataFax Toolbox. The initial appearance of the **DFsetup** window for any study is shown below. The first time this tool is started for a new study it will create an initial study setup in your workstation's memory. This initial setup automatically defines plate 501 which is used in all DataFax studies for the standard DataFax Quality Control Reports. The only variable on this page is the QC report ID number and it is defined automatically.

Figure 2.1. The Quality Control Report plate 501 is automatically created the first time DFsetup is started for a new study.



The top of the **DFsetup** window has the standard Motif menu bar. The bottom of the window contains a page counter, page turning buttons, and a message area. The majority of the tool layout is comprised of a scrolling window displaying the current CRF plate. There is insufficient screen dimension to display an entire 8.5 inch by 11.0 inch page at 100 dpi resolution. However, most of the page is visible and it can be scrolled, using the scrollbars on the right and bottom edges.

2.5. Importing CRFs

Before you can define variables you must import a rasterized version of each unique plate contained in the study CRFs. These images serve as background templates which are used to identify the location of the data fields on each CRF plate. DataFax permits study CRFs to be imported in 2 formats:

- PostScript
- PDF

Most desktop publishing packages support PDF and/or PostScript as an output format. Simply saving the case record book from the desktop publisher's format into either PDF or PostScript format makes the CRFs available for import. A single file can contain all of the CRFs, or multiple files can each contain a subset of the CRFs. This allows you to import them all at the same time, or only a subset, with one import command. If the PostScript or PDF file contains more than one copy of a CRF plate, the first copy encountered becomes the background template for that plate.

If PostScript is used, the maximum resolution **DFsetup** can handle is 1200 dpi.

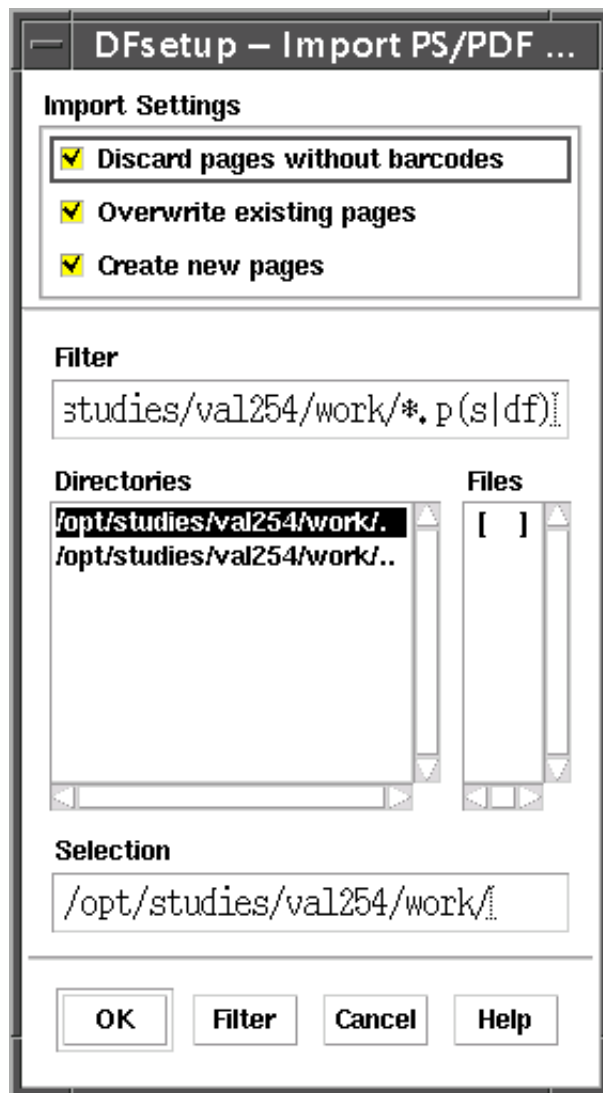
Both US letter and A4 CRFs can be imported as plate backgrounds in **DFsetup**. The setup tool is sensitive to page height and can determine whether or not the imported page is US letter or the slightly longer A4 size. If the page is A4, an adjustment will be made by the validation tool so that the DataFax record status field does not overlay user-defined data fields that may be positioned close to the bottom of the A4 CRF page.

2.5.1. Importing a PostScript/PDF Document into DFsetup

1. Choose Import PS/PDF from the Study menu.

The following dialog box will be displayed.

Figure 2.2. Import PS/PDF dialog box.



2. Navigate through the file selection box to find the PostScript or PDF file you want to import.
3. Choose the import options that you wish to use.

Check **Discard pages without barcodes** to discard all imported pages that do not have a DataFax signature line and barcode. This is a useful option when the PostScript or PDF document contains title and instruction pages in addition to barcoded CRF plates. If this option is not checked, you will be prompted for a plate number each time a page without a barcode is encountered during the import.

Check **Overwrite existing pages** to replace CRF plate images that are already defined in your setup with the current version of the plates in the PostScript or PDF document. Use this feature when you have made modifications to some page(s) in your desktop publishing package after defining them in setup. When this option is checked, the old variable definitions will not be lost; only the background raster image of the plate will be replaced. You can then edit the page to reposition, modify or add variables.

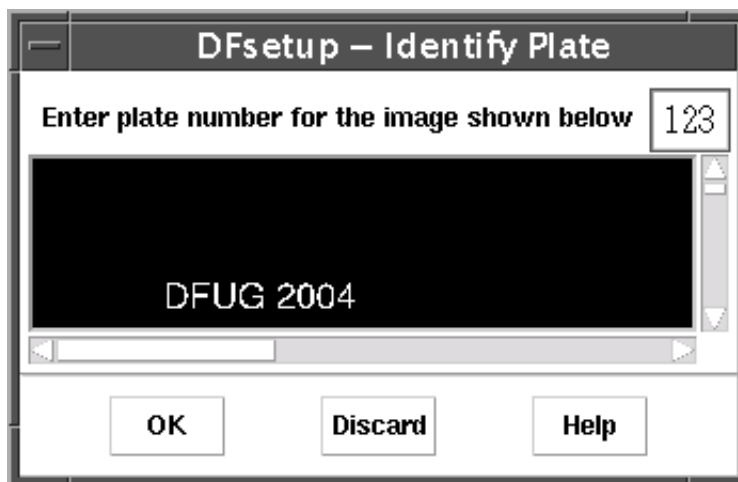
Check Create new pages to import all CRF plates in the PostScript or PDF document that have not yet been previously defined in setup.

4. Click OK to start the import operation.

This operation will take anywhere from a few seconds to several minutes depending upon the number of pages defined in the PostScript or PDF file.

If you did not check the Discard pages without barcodes option, you will be asked to identify each imported page that does not have a DataFax barcode. In the Identify Plate dialog you will be presented with a scaled-down image of the page in question and a location to enter the plate number. Text set in small type may be difficult to read when scaled down (as in the following example). However, it is usually not difficult to identify the page, particularly if you have a printed copy of the study CRF to refer to.

Figure 2.3. The Identify Plate dialog box.



Enter the appropriate plate number and click OK if you want to use the page in your setup; otherwise, click Cancel.

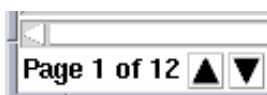
As each page is imported, the page numbering in the menu stripe will be updated to reflect the new page count.

2.6. Moving Between Pages

2.6.1. The Current Page and Total Pages Counter

The page that is displayed on the workstation screen is called the current page. The current page number and the total number of pages that have been imported into setup are displayed in the bottom-left corner of the setup window.

Figure 2.4. The current page and total pages counter



There are several methods of moving through the pages in your study setup.

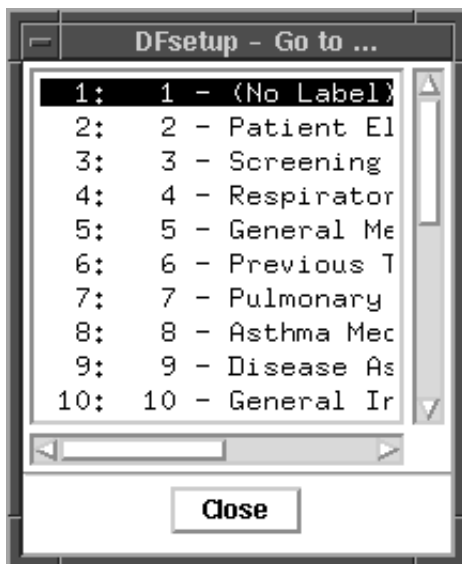
2.6.2. Moving to the Previous and Next Pages

- **Previous page.** Choose Previous from the Plate menu to go to the previous page. If the current page is page 1, the previous page is the last page of the setup. If you have audible bell enabled in your window environment, you will hear a bell when moving from page 1 to the last page.
- **Next page.** Choose Next from the Plate menu to go to the next page. If the current page is the last page you will return to page 1, and again, if you have audible bell enabled in your window environment you will hear a bell.

2.6.3. Go To an Arbitrary Page

- **Arbitrary page.** Choose Goto from the Plate menu to jump to a specified page.

Figure 2.5. The Plate - > Goto dialog box



2.7. Editing Page Definitions

After the import step has completed, you must edit each new imported CRF plate to specify a descriptive text label, how sequence numbering is determined, whether or not ICR is to be performed, and what additional processing, if any, might be necessary when a copy of the plate is received by fax.

To add or modify these attributes on the current page displayed in the **DFsetup** window, proceed as follows:

1. Choose Edit from the Plate menu.

The following Edit dialog box will be displayed:

Figure 2.6. The Edit dialog box.

2. Complete the appropriate fields in the dialog box.

Label. The Label field provides a descriptive text string up to 80 characters in length. While a label is not required, it is used by other DataFax programs and hence should be specified. **DFrouter** provides page labels to help in identifying mis-routed pages, and the **DFsetup** allows pages to be imported from other study setups by referencing plate labels.

Sequence Number. If the bar code across the top of the page includes the sequence number, select In the bar code, otherwise select In the first data field.

Perform ICR on plate contents. If you want the DataFax Intelligent Character Recognition (ICR) software to perform upon the plate's arrival in the DataFax database, select Yes. The ICR will attempt to read all types of data fields, with the exception of string fields.

Plate triggers early termination. If the plate belongs to a termination visit or cycle such as a death report, study termination visit or lost to follow-up assessment, select Yes, otherwise select No. Plates that signal early termination do so whether they exist as 'primary' or 'lost' records in the database.

Plate Triggered Procedures. Sometimes it is useful to have something happen automatically whenever a particular CRF plate arrives. For example, you might want to have adverse event reports printed out as they are received, or you might want them to automatically generate an email message to the adverse event monitor, or both. This is accomplished by specifying pre and post processes to be executed before and after the page has been read by the ICR software, respectively.

The ICRed data record and the CRF image file are both available, as arguments that you can pass to your procedure, as `$data` and `$image` respectively. When executing a Preprocess `$data` only contains the leading data fields through to the last bar coded field (which will be either the plate or sequence number).

To specify a procedure, enter the full command line to be executed in the Preprocess or Postprocess field, as appropriate.

Examples of plate triggered procedures can be found in Examples of Plate Triggered Procedures.



Warning

Any plate that enters **DFrouter** will not be handled by a plate triggered procedure until it is identified and forwarded to the appropriate study.

3. Click OK or Cancel.

Click Apply or OK to apply your edits to the current page, or Cancel to discard them. Move to the next page and repeat these steps. Continue until Plate > Edit has been completed for all of the imported pages.

Plate triggered procedures are also executed when data records are successfully imported into the study new record queue using **DFimport.rpc** with the `-n` option, unless they are suppressed with the `-s` option. If both a pre- and post-process have been defined in **DFsetup**, the pre-process is executed first, followed by the post-process. The values of `$image` and `$data` are the same in both processes. Failure of the pre-process does not prevent the execution of the post-process. If either pre- or post-process fails, a warning is written to the DataFax error log.

2.7.1. Examples of Plate Triggered Procedures

All commands used in plate triggered procedures must explicitly define all pathnames. This is because the commands are executed on the user's behalf by another program that is not aware of the user's shell prompt environment. The following examples assume that the DataFax installation directory is `/opt/datafax`, but this should be replaced with the correct pathname for your DataFax installation directory if it is not `/opt/datafax` on your system.

Example 2.1. Print the fax image of a CRF plate upon its arrival in the database.

```
/opt/datafax/lib/DFprint_filter $image
```

Example 2.2. Send an email message to username dan upon the arrival of a CRF plate in the database.

```
/bin/echo "A Serious AE form has arrived for study 254" | /bin/mail dan@cdsi.com
```

Example 2.3. Forward a CRF image to a fax machine having a fax number of 1-905-522-7284.

```
/opt/datafax/bin/DFsendfax $image 1-905-522-7284
```

Example 2.4. Forward a CRF image to the fax number 1-905-522-7284 and send user dan an email message warning him to expect the fax.

```
/opt/datafax/bin/DFsendfax $image 1-905-522-7284 ;  
/bin/echo "Expect a Serious AE form on your fax machine." | /bin/mail dan@cdsi.com
```

The line break has been added for readability. More than one UNIX command can be issued by separating each command with a semicolon (;).

For information about the **DFsendfax** and **DFprint_filter** utilities, please refer to *DataFax Programmer Guide*, *DFsendfax* and *DataFax Programmer Guide*, *DFprint_filter*.

2.8. Reordering Pages

New pages are created in a setup save file in the order in which they were imported. If you imported a PostScript or PDF file containing all study CRFs in which some plates were repeated (e.g. because they are completed at more than one visit) a new setup page will be constructed using the background image from the first copy of each new CRF plate number encountered. If CRF plates were not ordered sequentially by plate number in the imported file then the setup pages will not be in numerical plate order either. You can put them in plate number order by choosing Order from the Plate menu. This is not required, and is the only page ordering option provided. The reordering occurs as soon as you select > Order and it cannot be undone. Re-ordering plates may be helpful but it is not a necessary step.



Important

There is an implicit assumption in DataFax that pages within a case record book are sorted first by visit/sequence number and then by plate number within visit. However, this relies on the user barcoding the CRFs using ascending plate numbers within visit. For example, page 2 of a given visit has a higher plate number than page 1. Most of the time this holds true, but not always. If this does not hold true, plates will be retrieved in the Validation Tool (**DFvalidate**) according to ascending plate order (within visit) and not by order of pages within the case record book. Reordering plates within **DFsetup** will have no effect on the sorting and retrieval of CRFs in **DFvalidate**, i.e. the default order of ascending plate numbers within visits will be in effect.

2.9. Saving Your Work

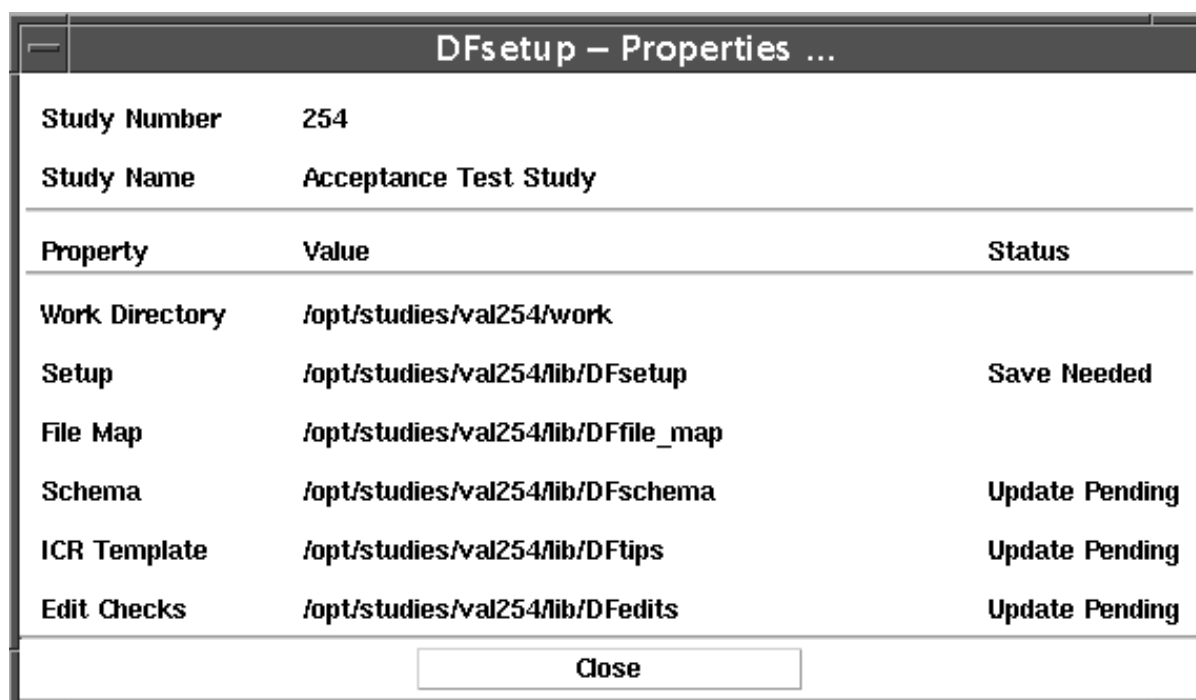
At this point, you have imported the study CRFs from your desktop publishing package into **DFsetup**. Each page with a unique plate number has created a new setup page. For each of these setup pages, you have defined a text label for the page, and a few other characteristics using the Plate > Edit feature.

The next step is to define the variables on each page. But before proceeding, now would be a good time to save the work you have done in **DFsetup** so far. The location of the setup save file for your study (and several other study-specific setup files) will have been defined in advance by your DataFax system administrator. You can review these specifications by reviewing the **DFsetup** properties as discussed in The Study Properties Window section below.

2.9.1. The Study Properties Window

1. Choose Properties from the File menu.

Figure 2.7. The Properties dialog box.



The **DFsetup** Properties window lists the location of all files used by **DFsetup**. It is a non-blocking window so you can keep it open while working in other windows if you wish. Notice that the location for the Setup save file in the above example is: `/opt/studies/val254/lib/DFsetup` and it's status is Save Needed, meaning that we have made additions and/or modifications which have not yet been saved.

2. Choose Save from the File menu.

This will actually cause your work to be saved to the named setup file. If your Study Properties window is still visible, you may notice the status change from Save Needed to Updating while the file is being written. When the save has completed, the status message will be cleared.

You should save your work at least every 10 to 15 minutes. In addition you can set Auto Save intervals as described in Backups and Auto Saves.

2.10. Quitting DFsetup

If you wish to exit from **DFsetup** at this time (or any other time), choose Exit from the File menu. If there are unsaved changes, you will be asked to confirm that these changes should be saved.

2.11. DataFax Variables

You will use **DFsetup** to define how each data field on each CRF plate becomes a data field in your study database. **DFsetup** includes features which can substantially reduce the amount of work you need to do in defining variables. One of the most useful is the ability to create variable styles which allow you to pre-define the constant characteristics of variables which have a common format. You can also copy variable definitions from other locations in the same study setup or from other study setups.

Each variable definition includes 3 attributes which must be defined:

1. data type
2. variable name
3. textual description.

Other attributes, including legal values, help information, formatting, coding, variable skipping rules, and those features which are specific to each data type are optional.

DataFax variable attributes are organized under 3 categories:

1. Basic Features
2. Specific Features/Coding
3. Edit Checks.

In this section, we will describe all of the attributes under each of these 3 categories. The sections that follow describe how to set these attributes using **DFsetup**.

2.11.1. Basic Features

2.11.1.1. Data Types

DataFax supports 6 data types:

1. number
2. date
3. string
4. choice
5. check
6. visual analog scale (VAS).

- **Number.** The numeric type defines both integer and fixed-point real numbers. Numbers may be signed (i.e. negative) or unsigned. Signed numbers must be specified in their format (see Format). Floating-point real numbers and numbers in exponential notation must be captured as string data types. The largest permitted value for a numeric field is the maximum 32-bit unsigned number, or 2147483647. Data fields based on this data type will be read by DataFax's ICR program.
- **Date.** The date type defines calendar dates in year, month, day format. The ordering of year, month, day can be specified and all permutations are permitted. The default permutation is year, month, day. 'Day' is represented by a 2-digit number. 'Month' is represented by either a 2-digit number or a 3-character string. 'Year' is represented by either a 2-digit or a 4-digit number. Date fields consisting entirely of numeric components as well as those containing 3-character months, will be read by DataFax's ICR program. Allowable date formats are described in Dates of the Specific Features/Coding section of this chapter.
- **String.** String data fields capture alphanumeric data as ASCII text. They are commonly used for comments, drug names, descriptive fields, etc. Such data fields will *not* be read by the ICR program. String data fields can also be used for dates and numbers that do not fit the DataFax style for these fields. Such fields must contain digits only (0-9) in a single rectangular box. In this case they are subsequently treated as numbers for purposes of legal range and edit check testing.
- **Choice.** The choice data type is used for response fields in which there are several possible choices from which exactly one choice must be selected (by checking or otherwise marking inside a box). Choice data types also support the notion of no choice having been made, although it is recommended that this be handled by an additional choice box labeled 'None' or 'Blank'. If multiple selections are allowed, the response field cannot have a choice data type, but instead must be multiple check fields.

Each possible choice must be defined with a unique, and definable, positive integer value together with a textual label. The choice data type can define at most 20 response options for one data field. Data fields based on this data type will be read by DataFax's ICR program.

- **Check.** A check data type is normally associated with a single box that is selected by checking inside the box. Check data types should not be used in 'Yes/No', or other paired response fields because it is difficult to determine if the field was actually considered when no selection has been made. Check data types are most appropriate for questions where the instruction is "Check all that apply". By default, variables defined with the check data type are coded 1 if checked and 0 if left blank, in the study database. However, users may choose their own numeric codes to represent "Checked" and "Blank", provided codes are integers within the range of 0-65535. Each check field must be defined with a code and textual label. Data fields based on this data type will be read by the ICR program.
- **VAS.** Visual analog scales must have the following attributes on DataFax case report forms:
 - Single horizontal line
 - Point size for the line must be in the range 0.5pt to 1pt, preferably 0.5pt.
 - The two endpoints of the line may each be marked with a short 0.125 inch vertical line if desired.

Optional anchors may be added anywhere along the line using a short vertical mark. For example, an anchor might be included to mark the indifference point at the midpoint of an Agree/Disagree scale.

Figure 2.8. Some visual analog scales created in FrameMaker®.



If you have a response field on a CRF that does not obviously match one of the six DataFax data types, define it as a string data type .

2.11.1.2. Variable Names

DataFax requires that each variable be named. We suggest that each variable name be unique although this is not a strict requirement.

DataFax supports two names for each variable:

1. Unique name
2. Generic name

The `Unique name` is required and is the name that is tested for uniqueness, while the `Generic name` is optional. A `Unique name` can be at most 80 characters in length and is made up of letters and digits although the first character must be a letter. You can also use the underscore (`_`) in any character position but the first; it is useful for improving the readability of long variable names. Upper and lower case letters are considered different when the name is tested for uniqueness. Choose variable names that are meaningful.

The `Generic name` has the same characteristics as the `Unique name` but it can be at most 16 characters in length, and it is not tested for uniqueness. Again, definition and use of the generic name is optional. Its use is most beneficial when exporting data fields to other programs that have more restrictive variable

names, for example SAS®.

2.11.1.3. Variable Descriptions

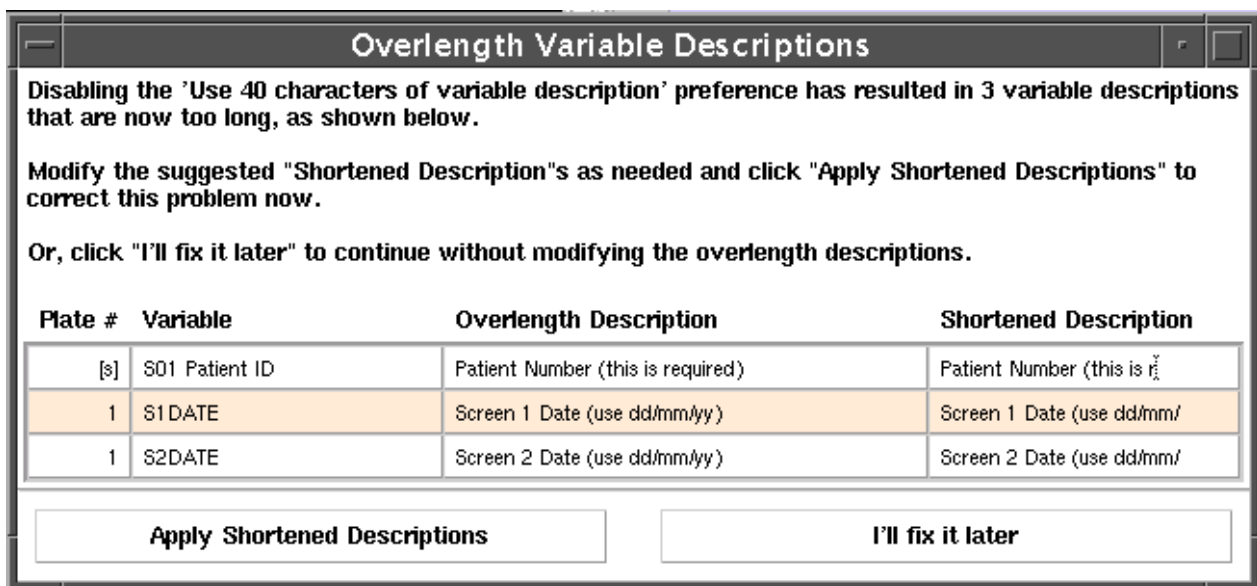
The variable description is required and can be at most 40 characters in length. This 40 character length can be specified by setting variable preferences. The study preference settings can be edited by selecting Preferences from the File menu. If 40 character length is not enabled, the default is 25 characters for the description. For more information on study preferences, see Set **DFsetup** Preferences.

The variable description is a simple text string and can be made up of letters, digits, spaces, and punctuation characters. The variable description that you provide is used on outgoing Quality Control Reports and hence the description should as efficiently as possible lead the recipient of these reports to the part of the CRF that is in question. Thus, the variable description should as closely as possible match the text that appears with the response field on the study CRF.

2.11.1.4. Overlength Variable Descriptions

Reducing the length of variable descriptions from 40 to 25 characters (using DFsetup - Preferences), will result in overlength variable descriptions for fields already defined with descriptions longer than 25 characters. If this occurs, the Overlength Variable Description dialog will be displayed, immediately upon applying the change to variable description length in the DFsetup - Preferences dialog.

Figure 2.9. DFsetup's Overlength Variable Descriptions dialog box.



The Overlength Variable Description dialog displays each field for which an overlength description has been detected in spreadsheet format. Each row of the spreadsheet identifies a problem field by: plate number, unique variable name, original (overlength) description, and proposed shortened (truncated) description. If the overlength problem occurs in the variable style, then the plate number column displays [s] and the variable name is replaced by the style name. Each variable affected by the overlength style is not shown as resolving the problem with the style description will resolve the problem in each variable defined with that style.

The Overlength Variable Description dialog presents you with 3 ways of resolving the variable description length discrepancies:

1. **Accept the truncated variable descriptions.** DataFax automatically truncates overlength descriptions at 25 characters and displays the result in the Shortened Description column of the spreadsheet. You can choose to accept these shortened descriptions by simply selecting the Apply Shortened Descriptions button. This will apply shortened descriptions to all variables as listed, and close the dialog.
2. **Edit the truncated variable descriptions.** You can choose to edit some or all of the truncated variable descriptions. To do this, select the Shortened Description to be edited and remove/add the desired text. The Shortened Description field is the only editable field in the spreadsheet. After making the desired changes to all problem fields, select Apply Shortened Descriptions to apply all shortened descriptions as displayed and close the dialog.
3. **Fix the overlength variable descriptions later .** You may choose to do nothing with the overlength variable descriptions at the time of changing your Preference settings. Selecting I'll fix it later results in the overlength descriptions remaining in the setup definitions and the Overlength Variable Descriptions dialog will close. Don't forget to correct these later, otherwise overlength descriptions will be truncated on QC reports.

The Overlength Variable Descriptions dialog may be accessed at any time by opening the DFsetup - Preferences dialog and selecting the Check Overlength... button to the right of the 40 character variable description option. The Overlength Variable Descriptions dialog will open and display all overlength variable descriptions that currently exist in the study setup definitions. You can then proceed to correct the problems as described above. Overlength variable descriptions can also be detected by running DataFax report **DF_ICschema** from the **DFreports** tool.



Note

Selecting Apply Shortened Descriptions applies but does not save the shortened descriptions. As for all other setup changes, you must select Save from the File menu in **DFsetup** to make your changes permanent.

2.11.1.5. Legal Values

A legal value definition enumerates the data values that are considered legal for that variable. Legal value testing is first performed by the ICR software. Any fields which do not pass the legal value test are left blank so that they can be checked and perhaps completed during data validation. The second legal value check occurs in **DFvalidate**. If a value is entered which falls outside the legal value set, it will change to the illegal value color (red by default). Although an illegal value can be forced into the data record, data records containing illegal values cannot have their status set to 'clean'. They must be added to the database as 'dirty' records. The one exception occurs for users with the highest validation level (7). They can force record status to 'clean' even if the record contains fields which have illegal values.



Note

The only time that this does not hold true is when one or more of the key fields are illegal because they are blank. Regardless of validation level, no user is able to assign a record with missing keys a status of anything other than **error**.

If legal values include one of the range delimiters ~ , or a blank space in the format, the legal range value must be enclosed in double quotes). For example, a date having the format "mmm dd, yyyy" may have a legal range value of "JAN 01, 1990"~today.

For Patient Id variables, a legal range can be defined using the meta-word `$(ids)`. `$(ids)` evaluates to the concatenation of all patient id ranges defined for all clinical monitor sites in the centers database. See Meta-Words for information about DataFax meta-words.

2.11.1.6. Help Messages

A short (one line) help message can be specified for each variable. If a help message and legal values are both defined, the help message will be displayed in **DFvalidate**'s one-line help window (located between the fax image and data entry windows), whenever an illegal value is encountered. A default Help message may be specified when setting variable preferences as described in the Set **DFsetup** Preferences section of this chapter.



Note

If the meta-word `$(ids)` has been used in the legal range definition for patient id variables, the literal `$(ids)` will be reported in any default help messages that are generated by referencing the patient id legal range.

2.11.1.7. Essential/Required/Optional Value

The data value for the variable (when presented in **DFvalidate** or **iDataFax**) can be specified as one of:

- **Optional.** There are no requirements for the value. There may be a value or the variable may be blank. Optional variables that are blank are identified with yellow color coding.
- **Required.** A required variable must contain a value or a missing value code and may not be blank. Required variables that are blank are identified with red color coding (in effect, the blank value is illegal).
- **Essential.** An essential variable must contain a value and may not be blank nor may it contain a missing value code. Essential variables that are blank or contain a missing value code are identified with red color coding. In **DFvalidate** and **iDataFax** it is also not possible for the user to interactively assign a missing value code to an essential variable.

In summary, an Optional variable does not require a value, while both Required and Essential variables require an actual data value, and an Essential variable does not accept a missing value code as a data value.

Key fields are intrinsically essential. Although it may be possible through a variable style to make a key field appear as though it is optional or required, a key field will always behave as essential. This means that a key field can never be assigned a missing value code nor can it be left blank. A record with one or more key fields that fail these tests can only be assigned a record status of `error`.

2.11.1.8. Reason Level

When a change is made to a data value, it may be important to record a reason for the change. It is possible to enforce this by specifying a 'Reason Level' on a study-wide basis or at the individual variable level. The reason level is the minimum record validation level after which a value change requires a reason. For example, if the reason level for the `Age` variable is specified as 2, then any change to the value of age when the record has a validation level of 2 or higher, will require a reason. The actual reason is specified interactively in **DFvalidate** and in the batch input file for **DFbatch**.

Reason for data change requirements must first be specified in the **DFsetup** preferences by selecting the Global preferences setting (See Set **DFsetup** Preferences). Reasons for change may be required

- per variable
- always
- never

If per variable, the Reason Level (1-7) must be specified at either the style or variable level when defining field attributes.

Once a change is made to a data field that requires a reason, it will thereafter always require a reason for subsequent data value changes, independent of what the record's validation level is. This remains true even if the data field's reason level is reset to `None`.



Reason Level != Validation Level

The reason level is the same as the validation level of the data record. It is not the validation level that the user has specified for performing validation of the set. For example, if the record's validation level is 2, the reason level for the current field is 3, and the validation level for the set is 4, no reason will be required for changes to the current field since the reason level of 3 is still higher than the record's validation level of 2. However, once the record attains a validation level of 3 or higher, changes to the current field will require a reason.

If the value of reason level is set to `None` (the default) then changes to the field's data value may be made at any validation level without requiring a reason. Note that it is not possible to set a reason level of 0 at either variable, style, or study levels. As a result, data entry or deletion of records from the new queue (where the records have a validation level of 0) is always possible without a reason.

2.11.1.9. Hidden?

Beginning with release 3.8, a field may be flagged as hidden to **iDataFax** users. A hidden field is not visible in any of the **iDataFax** data, query, or reason views - the user cannot select it, alter it, or otherwise interact with the field in any way.

The hidden attribute should be enabled for fields that are needed for data management but are not part of the original CRF (or whatever the data collection tool is). A field that is used to store the MedDRA code for a term reported by the user is a good candidate where the hidden attribute should be enabled - the code field is not present on the CRF nor should the **iDataFax** user have access to it. A field whose value is computed by an edit check is another good candidate. In general, fields that are added to the setup that do not have an obvious parallel on the CRF should always be flagged as hidden. Fields for entering source data should be visible while fields for storing derived data should be hidden.



Important

The hidden attribute applies only to **iDataFax** - users with access to other data view tools such as **DFvalidate** and **DFexport** will always be able to see hidden fields.

2.11.2. Specific Features/Coding

Some attributes are only relevant to a specific data type (e.g. date formats, choice coding schemes, decimal formats). The type specific attributes available for each data type are described below.

2.11.2.1. Numbers

There are 2 type specific attributes for numerical data fields. Both are optional.

1. **Display and Store.** Display and Store allows you to control the number of digits that are stored in the database and displayed in the data entry window of **DFvalidate**. If Display and Store are left blank, **DFsetup** will calculate the length of the field from the number of boxes clicked and both display and store this number of digits. However, if you have not defined a numerical field in the standard way with quarter inch boxes, you will need to specify the maximum length of the numbers that can be entered into this field and stored in the study database.
2. **Formatting.** Formatting specifications can also be added to numeric data fields. This is most commonly used to insert decimal places in the appropriate position, as printed on the CRF.

2.11.2.2. Dates

Date variables are comprised of year, month and day components. These components may be used in several combinations and in any order and may use any other character(s) to delimit the components, or use no delimiter at all. The type specific options for dates thus provide for formatting specifications.

Allowable date formats are composed of the following components for day, month and year.

- **DD or dd.** This is interpreted as a 2-digit number. Upper case means day is explicitly defined (zeros are not permitted). Lower case means day is optional and must be zeros (00) to indicate unknown.
- **MM or mm.** This is interpreted as a 2-digit number. Upper case means month is explicitly defined (zeros are not permitted). Lower case means month is optional and must be zeros (00) to indicate unknown.
- **MMM or mmm.** This is interpreted as a 3-character string. Upper case means month is explicitly defined (blanks and zeros are not permitted). Lower case means month is optional and must be zeros (000) to indicate unknown.
- **yy or YY.** This is interpreted as a 2-digit number using a specified implied range. Upper case YY does NOT mean that zeros may be entered for unknown values as for month and day. YY and yy are treated as equivalent.
- **yyyy or YYYY.** This is interpreted as an explicitly defined 4-digit number; any implied range is irrelevant. YYYY does NOT mean that zeros may be entered for unknown values as for month and day. YYYY and yyyy are treated as equivalent.

Date components specified in lower case support partial date rounding as long as the date format is logical. Partial dates and the rounding options available with DataFax are described in Set partial dates.

Formats are specified by typing certain combinations of day, month and year in the desired order. Although date components can be specified in any order, there are restrictions on the combinations acceptable to DataFax. The following combinations are NOT acceptable:

- DD/mm/yy
- DD/mm/yyyy
- DD/mmm/yy
- DD/mmm/yyyy.

It is not logical to have an explicitly defined day (day is known) if month is optional (not known).

If a separation character (e.g. a '/') is desired it too is typed in the format specification. When included, the separation character will appear in the database, will be displayed in the data entry window of **DFvalidate** and will be printed on Quality Control reports, without it having to be explicitly entered on the study CRF or typed in by a data entry clerk. Any character(s) may be used to delimit the date components, including no delimiter at all. Examples of legal date formats are:

- dd/mm/yy
- MMM DD, yyyy
- yyyy:mm.dd.

Missing values can be recorded for day (dd), or month (mm, mmm) and day (dd), by entering zeros in the missing field(s). If the year value or the entire date is missing it can be left blank or marked as `Not Available (*)`. If desired, the format characters 'DD' and 'MM' can be used in place of the lower-case counterparts to indicate that '00' is not an allowable value in the days or months part of the field, respectively. Note that this implies the value for the year cannot be marked as missing with the value '00'.



Note

If the date formats 'dd', 'mm' or 'mmm' have been used, missing values must be entered as zeros when validating records in **DFvalidate**.

2.11.2.3. Text

Text specific attributes control the number of characters stored in the database and displayed in data entry windows, text formatting and case conversions. Setting the maximum length of a comment text field stored in the database to be greater than the number of characters displayed on the screen is quite common. DataFax data files are not fixed length. Instead, each data field is delimited by the | character and is just long enough to hold the data value or text string entered. Thus for example, you could capture comments which are 160 characters long, but only display 30 or 40 characters at a time. When the length of a text field exceeds the maximum Display value the text field entry window scrolls so that you can continue entering text.

DataFax allows some limited formatting of text strings. All input can be forced to upper or lower case and fixed text can be prepended or appended to fixed length text strings. As an example, a dollar sign could be prepended to a monetary field and drug dosage units (e.g. mg.) could be appended to a drug dose field.



Note

The | character can not be used in text strings because this special character is used as the delimiter between data fields in the study database. Consequently, DataFax blocks any attempt to enter this character into string fields.

String fields can also be defined as containing only the digits 0-9 within a single rectangular box designed for ICR processing. Such fields are treated as numbers, not like other text strings. For these fields the specified store length is used to define the number of digits in the string, and no formatting is allowed. To convert a string field definition to a string of ICRed digits set the Treat as option to Pre-printed Numerals. The digits can be hand-written but pre-printed numerals are read by the ICR software with greater accuracy.



Style Definition

There is no pre-defined 'Simple' style which has the Treat as option set to Pre-printed Numerals. To define one or more fields in this way it is necessary to first create a style with this option set.

2.11.2.4. Choice

Choice fields provide several response options of which only one can be checked. The response checked is entered into the database as a numerical value and an optional text label can be added to the study database schema. The numerical coded value corresponding to each response option can be any integer between 0 and 65535, however each choice box must have a unique code. The default code for 'No choice' is 0 but this can be overridden. The text label has a maximum length of 32 characters. Each choice field may have up to a maximum of 20 response options.

2.11.2.5. Check

Check fields have the same attributes as a choice field with a single choice. The Check On and Check Off response can each be coded (the default is 1 for on, and 0 for off) and labeled (the default is 'on' for on, and 'off' for off).

2.11.2.6. VAS

The type specific attributes of a visual analog scale are the precision of the scale in cm, and the reference values for the minimum and maximum endpoints of the scale.

Currently, the default fixes scale precision at 0 decimal places and value range at 0 to 100. However, any precision and minimum and maximum reference values may be specified. Negative values are acceptable.



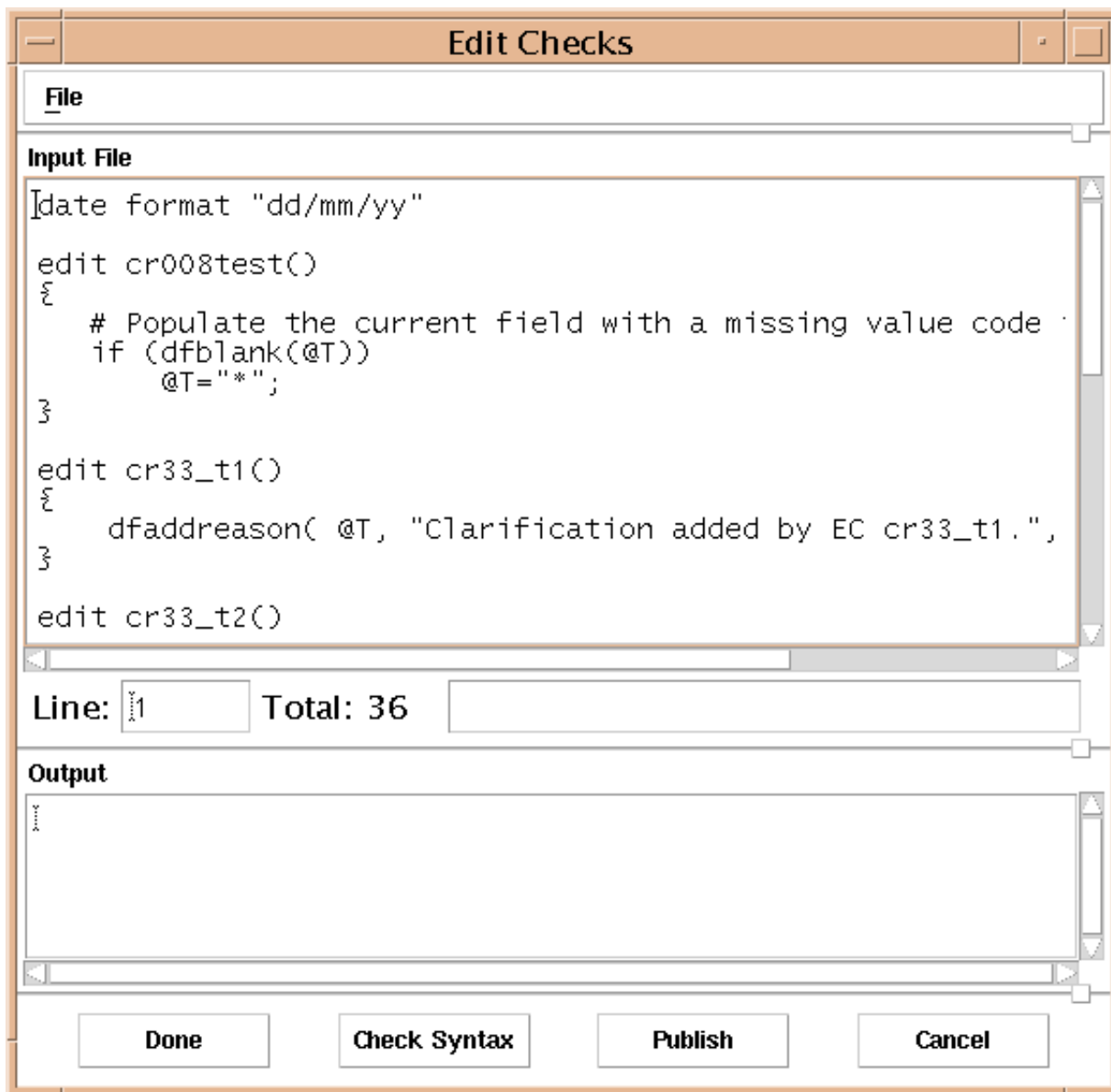
Note

Modifying a variable's definition once data exists in the database may result in a mismatch between an existing database value and its field definition. This may be particularly problematic if a variable's formatting and/or length has been changed. Although users will be notified of formatting or length inconsistencies during data entry in the **DFvalidate** tool (see *DataFax User Guide, Validation of Old Records Following a Setup Change*), care must be taken when editing any variable definition once a study is active.

2.11.3. Edit Checks

Edit checks are small programs, written in the DataFax edit checks language, that can be used to do things like calculate the value of a data field from other data fields in the database, perform integrity checks which involve referencing other variables, and add QC notes to the current data record. Edit check programs can be written in the edit checks input file as shown in the example below. This window can be opened by selecting Edit Checks from the Study pull-down menu.

Figure 2.10. DFsetup's Edit Checks dialog box.



Edit checks can be defined for each data variable on a form, if necessary. Within **DFsetup**, edit checks can be written to occur when the data record becomes current (on plate enter), when the data field becomes current (on field enter), when the data record loses currency (on plate exit), or when the data field loses currency (on field exit). At each of these points, zero or more edit checks can be defined to occur.

2.11.3.1. Edit Check Definition

The following procedure outlines the steps to be performed in defining and referencing edit checks in the Setup tool. These steps do not include a description of the edit check language used in programming the edit checks, which can be found in *DataFax Programmer Guide, Edit Checks*.

1. Open the Edit Checks editor.

Select the Edit Checks feature from the Study pull-down menu in **DFsetup** to open the Edit Checks editor.

2. Type each edit check program into the Edit Checks editor.

Each edit check program will begin with the name of the edit check, followed by the programming code that instructs the edit check what to do. Edit check naming conventions and language structure can be found in *DataFax Programmer Guide, Edit Checks*.

3. Check the syntax of the edit check program(s).

Before the edit check programs can be used, they must be syntactically correct. Syntax check the programs by choosing Check Syntax at the bottom of the dialog. Locate and correct reported errors before re-checking the syntax.

4. Optionally, publish the edit checks.

Edit checks are always executed in **DFvalidate** and **DFbatch**. To make them available for execution in **iDataFax**, they must first be published. For more information, see the following section, Section 2.11.3.2.

5. Locate the variable to which the edit check is to be attached.

Select the variable to make it the current variable. Open the Edit Checks section of the variable definition dialog.

6. Attach the required edit check(s) to the appropriate plate or variable.

There are 4 options for when the edit check is to be triggered:

- On plate enter
- On field enter
- On field exit
- On plate exit

Type the name of the edit check (names are case-sensitive) in the field next to the desired trigger definition. If the edit check definition includes parameters, ensure that runtime values for those parameters are specified at this point. It is legal to trigger the same edit check at different times.

More than one edit check may be triggered on the same variable by entering the names of all required edit checks next to the desired trigger definition. Use a comma (,) to separate each edit check name.

^[3] Where multiple edit checks are referenced by the same trigger, they are executed in the order that they appear in the list.

2.11.3.2. Publish Edit Checks

Edit checks are developed in **DFsetup** and then tested in **DFvalidate**. When they are ready to be deployed in **iDataFax** they must be published. This is accomplished by clicking the Publish in the edit checks dialog. Once edit checks are published, they become available to all subsequent **iDataFax** clients.

After new edit checks are written and tested, and/or existing edit checks are revised or deleted, it is again necessary to publish edit checks so that the updates are available in **iDataFax**.

**Note**

It is not possible to selectively publish a subset of edit checks. They must either all be published or not be published at all. To selectively control edit check behavior for **iDataFax**, an edit check can use the `dfTool` function to evaluate the context.

2.11.3.3. Edit Check Execution

Edit checks are executed interactively during field traversal in **DFvalidate** and **iDataFax**, and in batch in **DFbatch**. During field traversal, the edit check is executed according to the trigger defined in **DFsetup** (on plate enter, on field enter, on field exit, on plate exit). For example, if an edit check named `CheckAge` was defined on field exit from the `Age` variable, `CheckAge` would execute as soon as the user tabbed off the `Age` field.

The following behavior applies to interactive edit check execution.

- All plate enter edit checks will execute before any field enter, field exit and plate exit edit checks defined on that plate.
- All plate exit edit checks will be the last to execute on that plate.
- Field enter and field exit edit checks will execute in the order that the fields on the plate are traversed. If a field enter and a field exit edit check exist on the same field, the field enter edit check will be triggered first.

If multiple field enter or field exit edit checks are attached to the same variable, they will execute in the order that they have been entered in the On field enter or On field exit attribute of the variable's definition.

Edit checks may also be executed outside **DFvalidate** and **iDataFax** in batch mode (see *DataFax Programmer Guide, Batch Edit Checks*).

2.11.3.4. Skipping Irrelevant Variables During Data Entry

Sometimes response fields are positioned on CRFs in groups with one leading general question (e.g., Has patient been hospitalized since last visit?), followed by questions which ask for details (e.g., date, reason, outcome). The details are required only if the response to the general question makes them relevant. To allow you to skip irrelevant questions during data entry, **DFsetup** allows you to define variable skip rules. Each variable skip rule has two parts:

- the number of immediately following variables to skip
- values which cause the skip to occur

The value(s) in the trigger field which causes the skip to occur may be specified as one of the following:

- numeric code(s) defined in the trigger field definition (useful if the trigger field is type choice or check)
- a specific value entered into the data field (useful for numeric or string fields)
- the Meta-Words `$(blank)` or `$(legal)`

It is possible to define at most one skip specification for a variable. Cases where more than one specification is needed, for example, skip 2 fields if the response is "no" and 1 field if the response is "maybe", cannot be handled with this mechanism but instead can be handled in an edit check.



Skipping fields with edit checks

It is important to remember that if an edit check is attached to a field, and during validation the field is skipped as a result of a skip pattern, the edit check will not fire.

Fields which are skipped are handled in one of two ways depending upon the current validation level of the record:

- **Level 0 (record in new queue).** During new record entry, fields are skipped until at least one of the following conditions is true:
 1. the requested number of fields has been skipped, or
 2. the current field to be skipped is not blank.
- **Level 1-7 (record in database).** No change will be made to the field. If a value is in the field, it will remain in the field. This allows a data reviewer to record the responses that were on the CRF even though they may not have been expected.



Important

At no time will the data in a field be cleared as a result of a skip pattern.

2.11.3.4.1. Examples

Example 2.5. Skip condition with one value

This is the simplest case, where a number of fields should be skipped if there is a specific value in the current field.

Skip over fields
if value in

Example 2.6. Skip fields if value is blank

If one or more fields should be skipped if the value in the current field is not given, the meta-word \$(blank) can be used as follows:

Skip over fields
if value in

Note that it is not possible to test for the negation of this condition, that is, skip fields if the response is given (and hence is not blank). This can, however, be handled by an edit check.

Example 2.7. Skip conditions with more than one value

Consider the case where the `Other, specify` field for a `Race` choice question should be skipped if the response is not other. The CRF for such an arrangement might look like this:

Race	1	Caucasian
	2	African American
	3	Asian
	4	Other, specify: 12

If the responses for the `Race` variable are coded as shown (1=caucasian, ..., 4=other), then the specification to skip over 1 field if the response is 1, 2, or 3 would be:

Skip over	1	fields
if value in	1,2,3	

The condition in this case could also be written as 1-3.

2.11.4. Key Field Definition

The 4 key fields, study, plate, visit/sequence, and patient ID numbers appear on every CRF plate defined in the study database. While the study and plate numbers are fixed in the page barcode, the visit/sequence and patient ID numbers are not. Hence, you should be aware of several 'rules' for defining visit/sequence and patient ID fields during study setup. These are as follows:

- Key fields must contain only positive integer values and hence must be defined as either data type number, choice or check.
- Key fields must be essential.
- Key fields cannot be hidden.
- If the key field is defined as data type choice or check, all numeric codes must be unique and within the key's DataFax legal range.
- If a format is defined for the key field, it must not be longer than 5 digits for visit/sequence fields and 10 digits for patient ID fields. The format itself should only contain `ns` and optionally, 0-9 fixed digits.
- The store length for key fields must be between 1-5 for visit/sequence fields and 1-10 for patient ID fields. Display length should never exceed store length.
- Legal ranges, if specified, must fall in the DataFax legal range of 0-65535 for visit/sequence numbers and 0-2147483647 for patient IDs. If the DataFax meta-word `$(ids)` is used to define a legal range for the patient ID field, it must be used alone and cannot be combined with other legal value specifications. `$(ids)` is not a valid legal range specification for visit/sequence key fields.
- For patient ID fields, the format and/or legal range definitions must be the same across all plates in the study.

Executing the DataFax report program **DF_ICschema** from the Reports tool will detect any instances where the database setup does not conform to the above criteria.

2.12. Variable Styles

A variable style is a named specification that defines attributes of a variable definition that remain constant across all variables using that style. Examples of variables which share a common style include: patient ID numbers, dates, and questions which have the same response options (e.g. yes/no, mild/moderate/severe). By creating variable styles, and then referencing them as you proceed to define the variables on each CRF plate, you can both simplify the variable definition task and enforce consistency across vari-

ables which should have common attributes.

It is not necessary to define all styles before you begin to define variables. You can create and/or modify styles at any time, and style modifications will be automatically applied to all variables already defined using that style. It is however a good idea to both define styles and prepare an annotated copy of the study CRF which identifies which variables use them, before you begin to define variables using **DFsetup**. You should use the DataFax Study Setup Worksheets to aid you in these tasks.

2.12.1. The Style Catalog

The variable style catalog is part of the study schema and contains study-specific, pre-defined, `VisitDate` and `QC_Report_ID` styles.

2.12.1.1. Style Names

Each new style that you create is given a unique name which is added to the style catalog. Later you can retrieve styles by name from the catalog, for review or modification, or for application to a new variable definition. Following pre-defined `Simple` styles, variable styles are ordered alphabetically within the catalog. Style names may consist of letters and digits and an underscore '_' or space in any character position but the first, to improve readability.

2.12.1.2. Predefined `Simple` Styles

To get you started, the style catalog always includes the following styles:

- `SimpleNumber`
- `SimpleDate`
- `SimpleDateYYYY`
- `SimpleString`
- `SimpleChoice`
- `SimpleCheck`
- `SimpleVAS`
- `VisitDate`
- `QC_Report_ID`

The first 7 styles (each beginning with `Simple`) correspond to each of the 6 data types supported by DataFax. When you create a new style, you start with a copy of one of these `Simple` styles and then modify it to create the desired new study specific style. You must not delete any of these predefined DataFax styles.



Note

If you know that you will be using the `VALFMT` statement in **DFsas** to create SAS proc format statements for code labels during data extraction, you must make sure that you define both codes and labels in the style or variable definition. The style can be either a user-defined or DataFax `Simple` style.

2.12.1.3. `VisitDate` Style

The `VisitDate` style is used to define patient assessment dates, corresponding to each of the visits defined in the study visit map, and is used by the DataFax quality control programs. You will need to modify it to specify the date format used on your study CRFs. When defining new variables, the `VisitDate` style must only be used for date variables that correspond to study clinic visits or other patient

assessments as referenced in the study visit map (see *DataFax Study Planning Guide, Visit Dates*).

2.12.1.4. QC_Report_ID Style

Each DataFax quality control report is assigned a report ID number and printed on plate number 501. The `QC_Report_ID` style is used to define these QC Report ID numbers. This style assigns the variable name 'QCREP_ID' to both the `Unique name` and `Generic name` for report ID # (field #7) on plate 501. This is performed automatically by **DFsetup** the first time setup is performed on the study.

2.12.2. Creating a New Variable Style

To create a new style, choose `Style` from the `Variable` menu or choose `Variable Style` from the fast menu. The following non-blocking dialog box will appear next to your main setup window.

Figure 2.11. DFsetup's variable style definition dialog box

DFsetup – Style ...

Basic Features

Style: SimpleNumber

Delete: SimpleChoice, SimpleDate, SimpleDateYYYY, **SimpleNumber**

Type: Number ☐

Unique Name:

Generic Name:

Description:

Legal:

Help:

Value is: Optional ☐

Reason Level: None ☐

Hidden?: No ☐

Specific Features

Coding

Edit Checks

OK Apply Cancel

2.12.2.1. Defining Features

Whenever the variable Style dialog box is invoked, the Basic Features window, shown above, is presented. The variable Style dialog box has four separate windows: the Basic Features window, the Specific Features window, the Coding window, and the Edit Checks window.

Remember that those attributes you specify in a style cannot be changed in any variable defined using that style. Hence, only define those attributes of a style that will be constant across all variables which use it.

When creating a new style, the data type and a style name must be specified. All other variable attributes (Basic, Procedures and Specific) are optional and are only specified if they will remain constant across all variables which use that style.

2.12.2.2. Defining Styles vs. Defining Variables

The definition of variables is very similar to the definition of styles. In both cases you start by selecting an existing style from the style catalogue. To create a new style, modify the style attributes, give the style a new name and save it. When defining a new variable those attributes which have been predefined as part of the variable style will be grayed out and can not be changed. Those attributes which have not been specified in the style may then be set to complete the definition of the variable. The definition of styles and variables are both performed using the same Basic Features, Specific Features, Coding, and Edit Checks windows. Thus, we will describe how to use these dialog boxes to define style here and refer back to this section when we discuss the definition of variables.

To create a new Variable Style:

1. Choose a `Simple` style from the style catalog list.

The `Simple` style that you choose must have the same data type as the new style you intend to create.

2. Provide a new name in the Style field.

A unique variable name must be specified for each new style you create. This name will be added to the style catalog and used to define new variables based on this style.

3. Define Basic Features.

The figure below indicates those attributes that can be defined for a variable's Basic Features.

Figure 2.12. Defining Basic Features for a style named 'S01 Patient ID'.

DFsetup – Style ...

Basic Features

Style: S01 Patient ID

Delete: PWD Yes No, QC_Report_ID, QC_Report_ID10, **S01 Patient ID**

Type: Number ☐

Unique Name: ID\$(plate)

Generic Name: ID

Description: Patient Number

Legal: 1001~99999

Help: First 2 boxes = cent

Value is: Optional ☐

Reason Level: None ☐

Hidden?: No ☐

Specific Features

Coding

Edit Checks

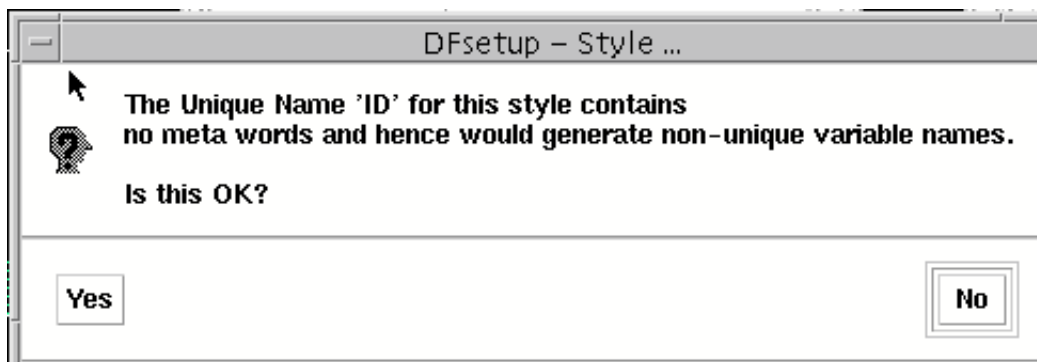
OK Apply Cancel

- Unique Name.** Each variable must have a Unique name (up to 80 characters long). Typically you will not complete the Unique name field in a style definition as you will want to provide unique variable names for each new variable that you create. However, in this example we have provided a name. The name itself is appended with the meta-word \$(plate) (see Meta-Words) which is evaluated to the number of the plate that the variable is defined on when the style is used to create a new variable. Hence, so long as there is only one variable referencing this style on each CRF plate, this Unique name will generate unique variable names like ID_001, ID_002,

etc.

If you intend to reference data fields by their variable names from other DataFax programs (like Query by Variable in **DFvalidate**), then it is a requirement that each variable have a **Unique name**. However, if you export your data and data dictionary to an external database package that does not have this requirement, you can complete the **Unique name** field in the style definition with a constant value. In general, you should make the **Unique name** unique and use the **Generic name**, defined below, for variables that will be exported. When you apply the style to the catalog, you will be warned about non-unique variable names with the following pop-up.

Figure 2.13. The warning dialog displayed as a result of non-unique variable names.



- **Generic Name.** **Generic** names (up to 16 characters long) are also typically specified when defining a variable, not when defining a style. These names are the names that are used to reference variables in the Edit Checks language and are also typically used when exporting the database to other systems (like SAS) that have different variable naming conventions than DataFax. In particular, DataFax allows the **Unique name** to be up to 80 characters in length, mixed case, and name uniqueness is enforced. A **Generic name**, on the other hand, can be up to only 16 characters in length and are not checked for uniqueness. Also, you may be exporting the data to an operating system like VMS that has upper-case naming conventions. Again, you should use the **Generic name** to enforce this naming convention.
- **Description.** Remember that the description field will be printed on DataFax Quality Control reports and thus should be as similar as possible to the text which describes the data field on the CRF. However, the format of QC reports limits the description to a maximum of 40 characters.
- **Legal.** Legal value testing is first performed by the ICR program and again during data validation. A legal value definition can include individual values, ranges of values, and any combination thereof in a comma delimited list. Value ranges are designated with a minimum value and a maximum value separated by the tilde (~) or dash (-) character. The minimum and maximum values are considered to be part of the legal range.

If you do not provide a legal value definition, all input for that variable is considered legal.

Example 2.8. Legal value definition that restricts values to 1, 2, or 9

Legal:

In most cases, it is not appropriate to specify legal value definitions for variables that have string data type. This is because inclusion testing is done by exact character comparison and range inclusion testing is done by ASCII collating sequence which is rarely what you would expect. An exception to this might be a variable assigned for collecting a unit of measure. This variable could be defined as a string and might have the following legal value definition.

Figure 2.14. A legal value definition for a string field.

Legal:

- **Help.** If a help message is specified it will appear in the one line message window (which separates the fax image and data entry windows in the Validation Tool) when you enter the data field. A useful help message is to just echo the legal value definition, as shown in the following example.

Figure 2.15. A help message consisting of a variable's legal value definition.

Help:

This is the default help message described in Set **DFsetup** Preferences later in this chapter.

- **Essential/Required/Optional Value.** The data value for the variable is one of Optional, Required, or Essential. If a data value is Required or Essential and the field is left blank, it will be coded with the illegal color (red) and the data record will not be allowed to have its record status set to `clean`. Similarly, a missing data value in an Essential field is also illegal and the data record cannot be assigned status `clean`.
- **Reason Level.** The data value can be tagged with a reason level. This is the minimum record validation level after which a change in the field's data value requires a reason.

If the reason level is set to `None` in a style, then it can be overridden at the field level. However, any other setting for reason level in a style, will require all fields using that style to have that same reason level.

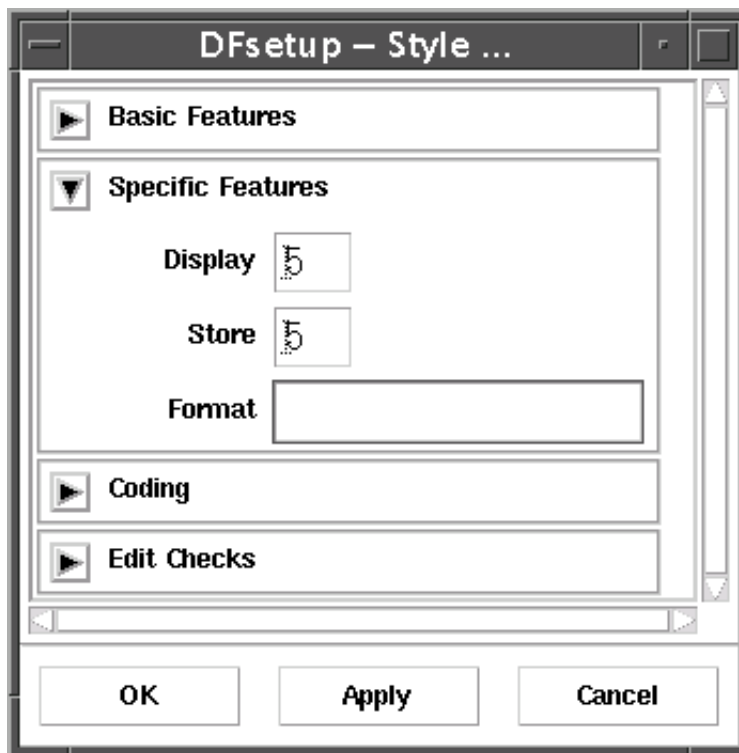
- **Hidden?** By default, fields are not hidden and this attribute is set to `No` - with this setting, the attribute can also be overridden at the field level. Setting the attribute value to `Yes` in a style will cause all fields using the style to be hidden from **iDataFax** users.

4. Define Specific Features

The definition of type specific features is optional. The Specific Features window varies, depending on which data type you indicated in the Basic Features window. The following examples show how to set Specific Features for each of the 6 data types. After entering your specifications, click OK or Apply to save them or Cancel to discard them.

- **Specific Features for Numbers.** For numbers the Specific Features window has the following appearance.

Figure 2.16. The Specific Features dialog box for numeric data types.



- **Display and Store.** For numeric data types, you can indicate the number of digits to display on the data entry screen and the maximum number of digits to store in the database. For numeric fields it is rare that these two numbers are different. If they are, the number of stored digits cannot be less than the number of displayed digits. If you leave these fields blank (or enter '0'), in the style and when creating a new variable with this style, then setup will calculate the number of digits from the number of boxes in the data field. However, if a numeric field is not laid out with boxes on the CRF then it will be necessary to complete Display and Store either in the style or when the individual variables of this type are defined. Thus for example, suppose you have left a line on which patient weight is to be entered in kilograms, and you plan to enter the values provided to a maximum of 2 decimal places. To allow for the rare individual who might be over 99 Kg you will want to allow for numbers like 104.55 and thus should specify Store = 5 digits. For numeric fields you will want to display all of the digits that are stored and thus will set Display to the same value as Store. The distinction between Display and Store is more relevant to text fields.
- **Format.** Format is used to add a fixed decimal point or units to a data field so that they need not be explicitly entered on the CRF. The following examples show how numeric data fields might be formatted. The character 'n' is used as a place holder for each digit which is entered on the study CRF. Any other characters will appear in their specified character position both in **DFvalidate** and in the value entered into the database.

The following example shows a data field for weight as it might appear on the CRF and the corresponding format. If no formatting was provided this number would be read and stored as a 4 digit integer. The Format specification (nnn.n) will insert the decimal place in the appropriate location.

Figure 2.17. Formatting specifications for a numeric field containing one decimal place.

Display:

Store:

Format:

If you want to allow a signed value in the database, you must use an 'S' or 's' in the field's format in place of an 'n' placeholder. For example, **snn** would be the correct format for a 2-digit number with a required sign (+ or -) as the leading character (i.e. -40). **snn** would be the correct format for a 2-digit signed number or a 3-digit unsigned number (i.e. -40 or 140). The lower-case 's' indicates that the sign (+ or -) is optional. When formatting signed numbers in **DFsetup** be sure to specify a correct legal range and Display/Store length for the field(s).

Formatting specifications can also be used to insert digits which are not entered on the CRF, as shown in the following example. Note that the values entered for Display and Store equal the number of digits actually entered in this data field. The zeros are not counted. However, they will be added and will appear both in the database and on the data entry screens.

Figure 2.18. Formatting specifications for a numeric field containing fixed digits.

Display:

Store:

Format:

- **Specific Features for Dates.** The Specific Features window for dates is shown in the following example. The appearance of the Specific Features window is the same for both the `SimpleDate` and `SimpleDateYYYY` styles.

Figure 2.19. The Specific Features dialog box for date data types.

The screenshot shows the 'DFsetup - Style ...' dialog box. The 'Basic Features' section is collapsed. The 'Specific Features' section is expanded, showing the following settings:

- Display:** 6
- Store:** 6
- Format:** dd/mm/yy
- Range for 2 digit years is:** 1920 - 2019
 - ☐ by inheritance
- Set partial dates:** never
 - ☐ by inheritance

Below the 'Specific Features' section are two collapsed sections: 'Coding' and 'Edit Checks'. At the bottom of the dialog are three buttons: 'OK', 'Apply', and 'Cancel'.

- **Display and Store.** Store will range from 6 to 9 characters depending on the format of month and year components used. Delimiters (i.e., /), are not counted as characters. You will normally want to have Display equal to the Store value so that the entire date will be visible on data entry screens. However, if screen space is tight, you might set Display to a lower number, for example 4. This would create a scrolling window for dates with only 4 digits visible at a time. On entering such fields only the last 4 digits would be visible.
- **Format.** The formats used throughout the study CRFs for dates must be defined in the date Specific Features window. Month, day and year are specified in date formats in the same order as they appear on the CRF. If the day and month parts of the date are not allowed to take on the value '00', then the formats DD and MM or MMM should be used, respectively, instead of dd and mm or mmm. If dd, mm or mmm are used and values are not known, zeros must be entered as place holders during data entry in **DFvalidate**. An optional single character delimiter may be included to separate the 3 parts of each date. No delimiter is also acceptable. The following example shows how a date field might appear on a CRF and an appropriate date Format definition. Other legal formats for this field might include: dd:mm:yyyy and dd mm, yyyy. Remember that the | character can not be used as a delimiter. This character is reserved by DataFax as the database field delimiter.

Figure 2.20. Formatting specifications for a date field containing a 2-digit day and month, and a 4-digit year component.

Display:

Store:

Format:

- **Range for 2 digit years.** This feature allows specification of an implicit range for dates formatted with a 2 digit year. The implied range must cover a consecutive 100-year period and can be specified by completing the year which begins the range. By default, the implied range assumed by DataFax is 1920~2019 if a range is not specified. For example, an implied range of 1910~2009 for a 2 digit year (format `yy` or `YY`), will indicate that the year is to be interpreted as falling within the specified range of 1910~2009. Implied ranges also apply to legal range specifications. For example, a legal range of 10/01/94~01/01/05 specified for a 2 digit year having an implied range of 1910~2009, will result in a legal range of 10/01/1994~01/01/2005.



Note

The Range for 2 digit years may be defined on a study wide basis (see Set **DFsetup** Preferences), at the date style level, and at the individual variable level. If by inheritance is checked, the implied range at the style level will be ignored and the range defined on a study wide basis will be used as the implied range for the style.

Implied range specifications will be ignored for dates defined with a 4 digit year (format `yyyy`) as 4 digit years are interpreted as explicitly defined.

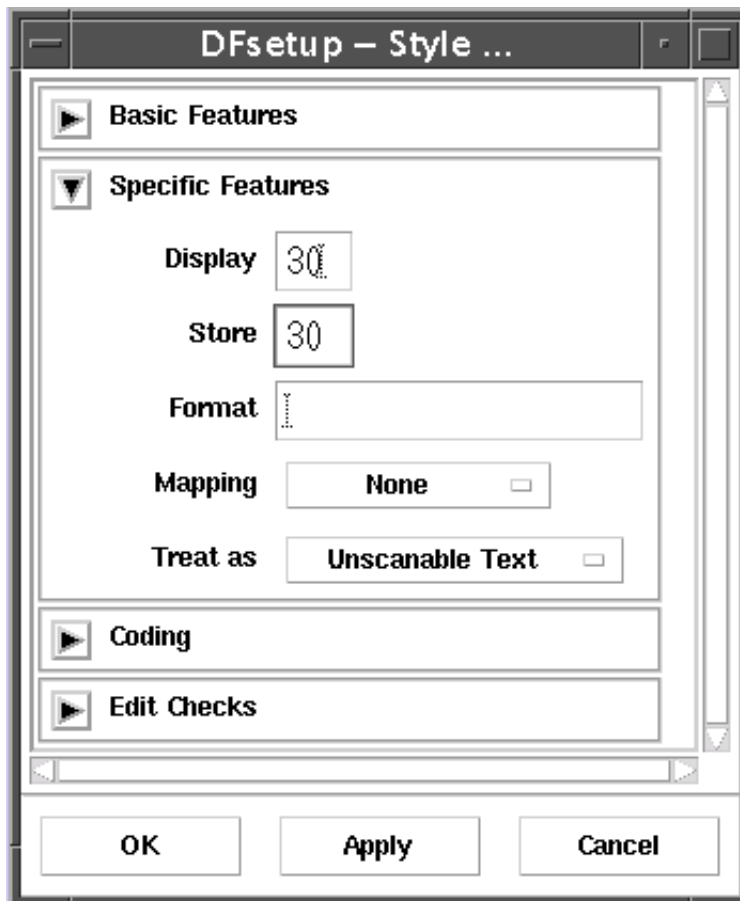
- **Set partial dates.** Dates may also have a rounding option to specify how partial dates (e.g., day missing or day and month missing) are to be treated in legal range tests. These options include:
 1. **never.** No rounding is specified (implemented by using `DD` and `MM` or `MMM` for day and month in the date format.
 2. **to first of month/year.** Implemented by using `dd` for day and `mm` or `mmm` for month in the date format. If `dd` and `mm` or `mmm` are used, zeros must be entered as placeholders during data entry.
 3. **to middle of month/year.** Implemented by using `dd` for day and `mm` or `mmm` for month in the date format. If `dd`, `mm` or `mmm` are used, zeros must be entered as placeholders during data entry.
 4. **to last of month/year.** Implemented by using `dd` for day and `mm` or `mmm` for month in the date format. If `dd`, `mm`, or `mmm` are used, zeros must be entered as placeholders during data entry.

Components of the date must be specified in lower case in order to support partial date rounding. Date components and formats are further described in Dates.

Set partial dates may be defined on a study wide basis (see Set **DFsetup** Preferences), at the date style level or at the individual variable style level. If by inheritance is checked, the rounding option set at the style level will be ignored and that rounding defined at the study level will be used as the 'set partial dates' specification.

- **Specific Features for Strings.** The Specific Features window for String variables is shown below.

Figure 2.21. The Specific Features dialog box for string data types.



- **Display and Store.** In the above example Display has been set to 30 characters and Store has been set to 30 characters. Display defines the number of characters visible in the data entry window in the Validation Tool and Store defines the maximum length of the field to be stored in the database. When Display is set less than Store the text field is displayed in a scrolling window on data entry screens. When the entered text exceeds the display length the text is scrolled left so that additional text can be entered.
- **Format.** Formatting is rarely specified for text fields but may be used to add constant units of measurement. In text formatting the ^ character is used as a place holder for text typed from the keyboard. Anything else is treated as fixed in the position indicated. If the fixed text appears at the beginning of the text field as in the first example below, a single ^ will serve as the place holder for any number of characters entered into the text field. However, when the fixed text appears at the end of the string as in the second example the fixed text will only become part of the data field when all ^ place holders have been filled. The following example shows a legal format for a monetary style in which the amounts will not exceed \$999,999.

Figure 2.22. Format specifications for a string field containing a constant unit of measure at the beginning of the text.

Display:

Store:

Format:

Mapping:

Treat as:

The following example of string formatting shows a dose field in which the units of measurement (mg) are to be included in the database, and the number of characters in the specification of dose will not exceed 4.

Figure 2.23. Format specifications for a string field containing a constant unit of measure at the end of the text.

Display:

Store:

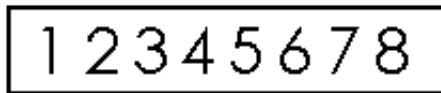
Format:

Mapping:

Treat as:

- **Mapping.** Text fields can be mapped to upper or lower case by selecting the desired option. When case mapping is set it occurs automatically when text is entered for that field in **DFvalidate**.
- **Treat as.** For text fields containing alphanumeric characters select Treat as unscannable text. String fields of this type are not read by the ICR software and are left blank for completion by data entry staff. For text fields consisting only of digits printed in a single rectangular box, select Pre-printed Numerals. These fields can be read by the ICR software. ICR accuracy is best when the digits are pre-printed using: Avant Garde-Book font, 18 point, with 15-20% spread. But it is also possible to use such fields for hand written numbers. The rectangular box printed on the CRF for these fields should be 1/3 inch high and just long enough to hold all of the digits without crowding. The digits, whether pre-printed or hand-written should be positioned mid-way between the upper and lower edge of the box and clearly separated from each other, as shown in the following example.

Figure 2.24. Pre-printed numerals in a string field on a CRF page.



ICR on such fields is much more accurate than for hand written digits.

- **Specific Features for Choice Variables.** There are no specific features for choice variables.
- **Specific Features for Check Variables.** There are no specific features for check variables.
- **Specific Features for VAS Variables.** There are three specific features for VAS variables. The features are:
 1. Precision of the scale in cm.
 2. Reference value for the leftmost end of the scale (the minimum).
 3. Reference value for the rightmost end of the scale (the maximum).

The values for all 3 may be changed from their default.

5. Define Coding

Coding is required for DataFax choice and check data fields but is optional for numeric data types.

- **Coding for Numeric Variables.** **DFsetup** allows you to associate up to 20 code-label pairs with a variable or variable style. Most numeric variables are not coded as the range of possible values is too great. However, if you define a numeric field for which only a few valid responses are accepted, these responses can be coded with meaningful labels. DataFax gives your SAS programmer access to these labels when data is exported from DataFax. Usually, fields for which only a few responses are valid are created as choice fields.
- **Coding for Choice Variables.** **DFsetup** allows you to define numerical codes and corresponding text labels for each of the response options provided for a choice field. The maximum number of response options for choice fields is 20 and thus space is provided to define 20 codes and labels, plus the code and label for 'no choice'. The default coding is to use the ordinal position of the response within the list of response options, and thus will be a number in the range 1-20. The default code for no choice is 0. The following example shows the dialog box for a choice style with 3 response options labeled **mild**, **moderate** and **severe**.

Figure 2.25. Coding specifications for a choice field containing 3 response options.

DFsetup – Style ...

Basic Features

Specific Features

Coding

4 leading chars of the Label should be used by the validation tool

Box	Code	Label
0.	0	none
1.	1	mild
2.	2	moderate
3.	3	severe

Edit Checks

OK Apply Cancel

The numbers at the far left of the window correspond to the order in which the choice options are laid out on the CRF. The first row, labeled 0 under the Box column, is for coding no choice and its label. You must ensure that you indicate choice values and codes in this same order as DataFax cannot infer this for you. This is a critical part of the setup and will affect the data translation if done incorrectly. When you subsequently define coded choice variables, be sure to examine the variable widget carefully as it will reflect both the ordering of the choices and their coded values. This is discussed further in The Variable Widget.

The Code column is used to specify the numerical value to be entered into the database when this option is selected. These values must lie in the range 0~65535. It is a requirement that each code be unique for a variable.

The Label column is used to specify a text label for each of the possible response options. These labels will be entered into the study database schema, for variables which are defined using this choice style.

**Note**

The default legal range for a choice field is `$(choices)` which evaluates to all of the defined choice codes including the 'no choice' code. If 'no choice' is not a legal response, then the legal range must be modified from the default to explicitly exclude 'no choice' or the setting for Value is: **Required/Optional** must be set to **Required**.

- **Coding for Check Variables.** **DFsetup** allows you to define the numeric codes and labels for the status **not checked** and **checked** for check fields. The default coding for **not checked** and **checked** are 0 and 1, respectively, and the corresponding labels are off and on.

The two codes must both be numeric values and must not be equal to each other.

**Note**

The default legal values for a check field include **checked** and **not checked**. If **not checked** is not a legal response, then the legal range must be modified from the default to include only the checked response or the setting for Value is: **Required/Optional** must be set to **Required**.

6. Apply or Cancel Your New Style Definition.

Select Apply to save your style definition or Cancel to exit without saving your definition.

2.13. Modifying an Existing Style

You can easily modify a style that is already defined in the style catalog. Although possible, it is recommended that you not modify the *Simple* DataFax supplied styles. Instead, create a new style based on the *Simple* style as previously described in *Creating a New Variable Style*. The usual exceptions to this rule are the *SimpleDate*, *SimpleDateYYYY* and *VisitDate* styles. The DataFax default format for dates is *yy/mm/dd* for *SimpleDate* and *VisitDate* styles and *yyyy/mm/dd* for *SimpleDateYYYY*. Retrieve the *SimpleDate* and *VisitDate* styles and modify Format under Specific Features if the format for your study dates is something different.

To modify an existing style:

1. Choose Style from the Variable menu.

This will open the Style dialog box.

2. Choose by name the style you want to modify from the style catalog scrolling list.

Styles are ordered alphabetically following the listing of the DataFax *Simple* styles.

3. Modify the desired features and/or add new features to the style.

Remember that each defined style attribute appears in every variable that references the style and hence each modification will automatically appear in all variables using that style. This applies to all variables already defined using that style as well as to all future variables defined with it. This is one of the real benefits of creating styles and using them to define variables.

However, if you define a new attribute for a style and that attribute has already been specified for some variables which use that style, then the new value of the style attribute will override these previous specifications. This wipes out anything you have entered in that attribute field for all variables using that style; so be sure that this is what you want before adding new attributes to an existing style. Before committing any modifications, you will be warned that your changes will override the existing style definitions.

4. Click OK or Apply to overwrite the existing style in the catalog.

If you decide not to keep the changes you made to the style, click Cancel.

2.14. Deleting a Style

A style can only be deleted if there are no variables referencing that style in the setup, as this would leave dangling references. If a style is in use you must first change the style of each of the variables that references it before the style can be deleted.

To delete a style:

1. Choose Style from the Variable menu.

This will open the Style dialog box.

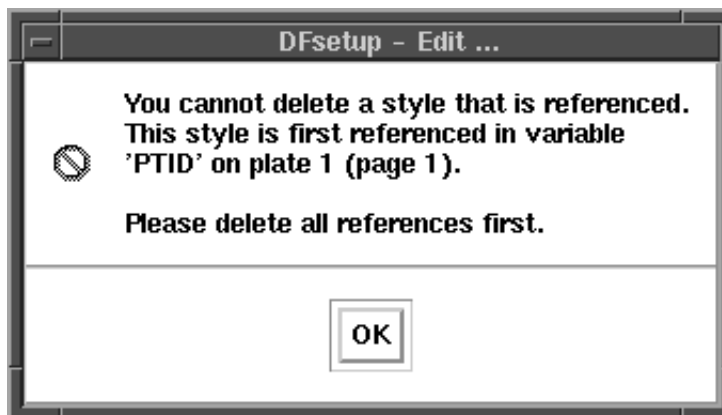
2. Choose by name the style you want to delete from the style catalog scrolling list.

Styles are ordered alphabetically following the listing of the DataFax Simple styles.

3. Click Delete to delete the style from the catalog.

If the style cannot be deleted because there are outstanding references to it, you will see a warning message like the one shown in the following example.

Figure 2.26. A warning will appear if a variable references a style that is marked for deletion.



Select OK and then resolve each of the outstanding references before re-attempting to delete the style.

2.15. Defining Variables

The procedure for defining variables is identical to that already described for defining variable styles with one important difference. Each variable definition only applies to a specific data field on the current setup page. Thus you must begin a variable definition by indicating the location of the data field on the CRF. This can be accomplished by clicking on each of the boxes that make up the data field (if they exist), or by dragging out a bounding box that defines the extent of the variable's location. After locating a variable, select New from the Variable to open the **DFsetup** New dialog box. Begin by choosing a style from the style catalog and then defining relevant attributes not already defined in the style. When complete, the variable specifications are applied to the in-memory version of your setup and a screen representation of the variable (referred to as a variable widget) will be drawn. This sequence of steps is repeated for each new variable that you need to define.

You can define new variables in whatever order you find most efficient. When all variables have been defined on the current page, you can reorder them in the order that you want the corresponding data entry fields to be completed, by choosing Order from the Variable menu.

2.15.1. Set *DFsetup* Preferences

There are a number of features that you should consider setting before you begin to define variables. They are used to set default specifications for study-wide (global) and variable settings, as well as **DFsetup** aids such as Guides and Color settings.

Choose Preferences from the File menu to display the Preferences dialog box which is shown below.

Figure 2.27. The Preferences dialog box allows specifications to be set for 4 areas of DFsetup; Global, Variables, Guides, Color.

The screenshot shows the 'DFsetup - Preferences' dialog box with the 'Global' tab selected. The 'Study Start Year' section has a text box containing '1990'. The 'Reason for Data Change' section has a checkbox labeled 'Reasons for data change are required' which is checked, and a dropdown menu set to 'per variable'. The 'Auto Save' section has a checkbox labeled 'Perform auto save every (minutes)' which is unchecked, and a text box below it showing 'Last auto save: (none)'. At the bottom are 'OK', 'Apply', and 'Cancel' buttons.

2.15.1.1. Global Settings

- **Study Start Year.** The study start year gets written to DFschema; and is used by DataFax Generic reports in time-based calculations. This setting has no implications on other DFschema; definitions such as legal range specifications.
- **Reason for Data Change.** This setting controls whether or not a reason is required for a change in a data field's value. There are 3 options to choose from.
 1. **per variable.** a reason for change is required as defined in the variable attribute of Reason Level for a given field. Hence, the reason for change requirement adheres to the individual field definition.
 2. **never.** a reason for change is not required for any field in the study that has "None" specified for its Reason Level attribute. The "never" specification can be overridden at the variable or style level, provided its Reason Level is set to a value other than "None".
 3. **always.** a reason for change is always required for each data change for any variable, independent of validation level or a variable's reason level. This setting does not apply to level 0 records (new records).
- **Auto Save.** This feature allows the user to enable/disable automatic saves of the study setup. See Backups and Auto Saves.

2.15.1.2. Variables Settings

- **Description.** This setting controls the number of characters that are to be used in each variable's description. The default is 25 characters unless the toggle button is enabled. If enabled, this setting also controls the length of labels specified in `DFpage_map`. The Q&A/Refax labels may be up to 32 characters and the Patient Status labels 18 characters if the toggle button is enabled, otherwise the default of 17 and 6 characters respectively, prevail.

The length of variable descriptions can be modified at any time during study setup. If description length is changed from 40 to 25 characters and existing variable descriptions have greater than 25 characters, you will be warned by the appearance of the Overlength Variable Descriptions dialog. This dialog can be used to fix the problem descriptions (see Overlength Variable Descriptions).

- **2 digit years.** As previously described in Range for 2 digit years, range for 2 digit years allows you to specify an implicit range for dates formatted with a 2 digit year. Specifying an implicit range using the variable options feature defines the date range on a study wide basis. The range that is specified at the study level may be overwritten at both the variable style and individual variable levels.
- **Set partial dates.** This feature allows you to specify how partial dates (i.e., day missing or day and month missing) are to be treated in legal range tests. For a complete description of rounding options for partial dates see Set partial dates. Specifying partial date rounding using the variable options feature will set rounding options on a study wide basis. Rounding options specified at this level may be overwritten both at the style and individual variable levels.
- **Auto Help.** If Auto Help is set to None, the help message field in the Basic Features window will be left blank unless you enter a help message yourself for each new variable you define. If you set Auto Help to Apply to New Variables, the default help message Legal values are: `$(legal)` will be automatically entered into the help field whenever you create a new variable, unless you override the default by specifying your own help message. If you set Auto Help to Apply to All Variables, the default help message will be added to any previously defined variables which currently have empty help message fields, as well as being used as the default message for any new variables you subsequently create.
- **Auto Unique Name.** If Auto Unique Name is set to None, the `Unique name` will be left blank unless you enter a variable name yourself for each new variable you define (remember that a variable name is required for each variable). If you set Auto Name to Apply to New Variables or Apply to All Variables, the default variable name, `p$(rplate)v$(field)`, will be automatically entered in the variable name field whenever you create a new variable. (See Meta-Words).
- **Auto Generic Name.** Auto Generic Name is similar to Auto Unique Name. If set to None, the `Generic name` field in the Basic Features window will be left blank unless you complete it yourself for each new variable you define. You can alternatively have **DFsetup** automatically construct a `Generic name` using up to 16 characters from the beginning of the variable `Unique name`. Recall that `Unique name` is a required field which must be completed in the Basic Features window for all variables. If Auto Generic Name is set to Apply to New Variables a `Generic name` will only be generated for any new variables you subsequently define. If you set Apply to All Variables a `Generic name` will also be generated for any previously defined variables which do not have a `Generic name`. Again, you can override the default by specifying a `Generic name` for any new variable you define.

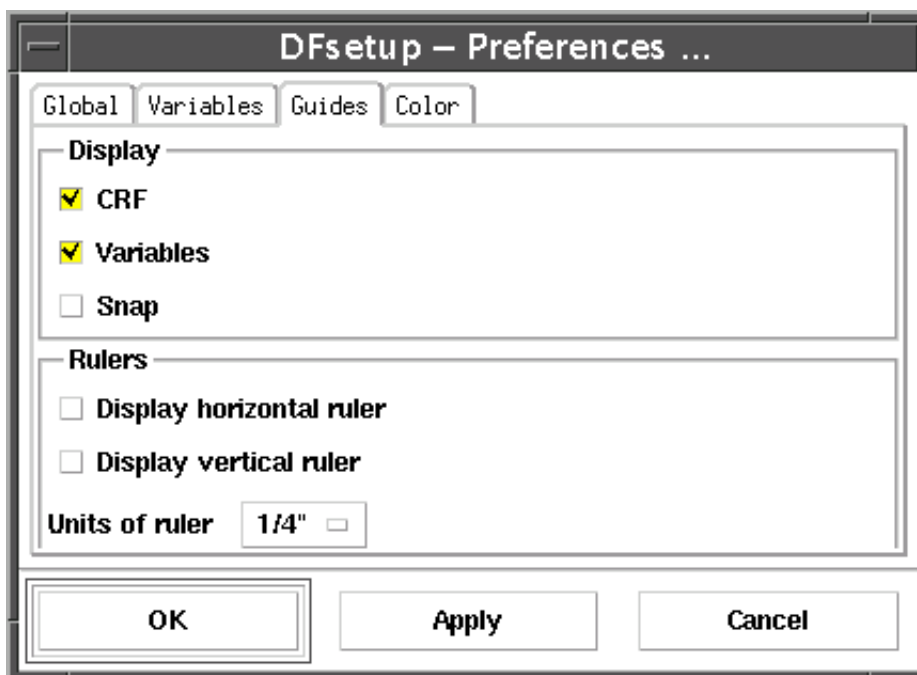
2.15.2. Guides Settings

2.15.2.1. Guides Menu

There are several guides which alter the appearance and function of the main **DFsetup** window. Some of the guides are available to help you position variables. Other guides toggle the visibility of setup features.

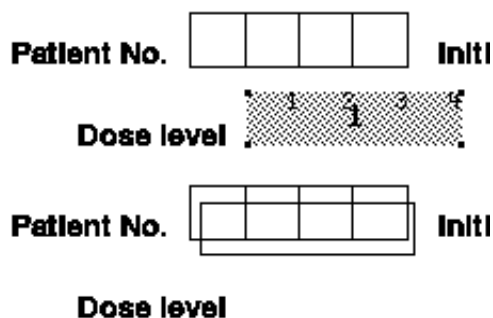
If a guide is toggled on, it is in use as in the example shown below.

Figure 2.28. The Guides dialog. The > Variables and CRF are in use.



- **CRF.** You can toggle the display of the underlying CRF image by choosing CRF from the Guides dialog. Under most circumstances you will want to have the CRF visible. In particular, the CRF image is an extremely useful guide for positioning variables.
- **Variables.** Variable widgets show you where variables have been defined on the current page. You can toggle the display of variable widgets (on/off), by choosing Variables from the Guides dialog. Under most circumstances you will want to have the variable widgets visible.
- **Snap.** Snap helps you position a variable over data boxes and lines on an imported CRF plate. When Snap is active, you need only drag and position a variable widget in the vicinity of the variable's actual location on the CRF. When you release the variable widget, **DFsetup** attempts to 'snap' it to the lower-left corner of the nearest data square on the CRF. The lower-left corner of the variable widget and its CRF counterpart must be within 0.08 inch of each other for the snap to succeed.

Figure 2.29. Three steps involved in 'snapping' a variable widget over its location on the CRF.



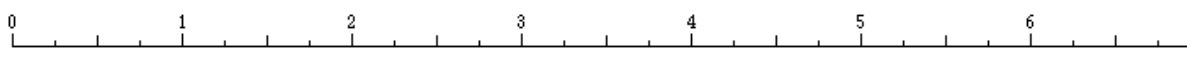
Patient No.  Initl

Dose level

It is not possible to position variable widgets outside the visible boundaries of the page by either dragging or snapping them into place. The Snap feature will ensure that the widget stays within the page boundaries for both US letter and A4 backgrounds in **DFsetup**.

- **Rulers.** Typically, you will position variables by their positions on an imported CRF page. Additionally, you can use the ruler guides on the setup page edges to position variables at specific horizontal and vertical locations. By default, both the horizontal and vertical rulers are displayed and the unit of measure is 0.25 inch.

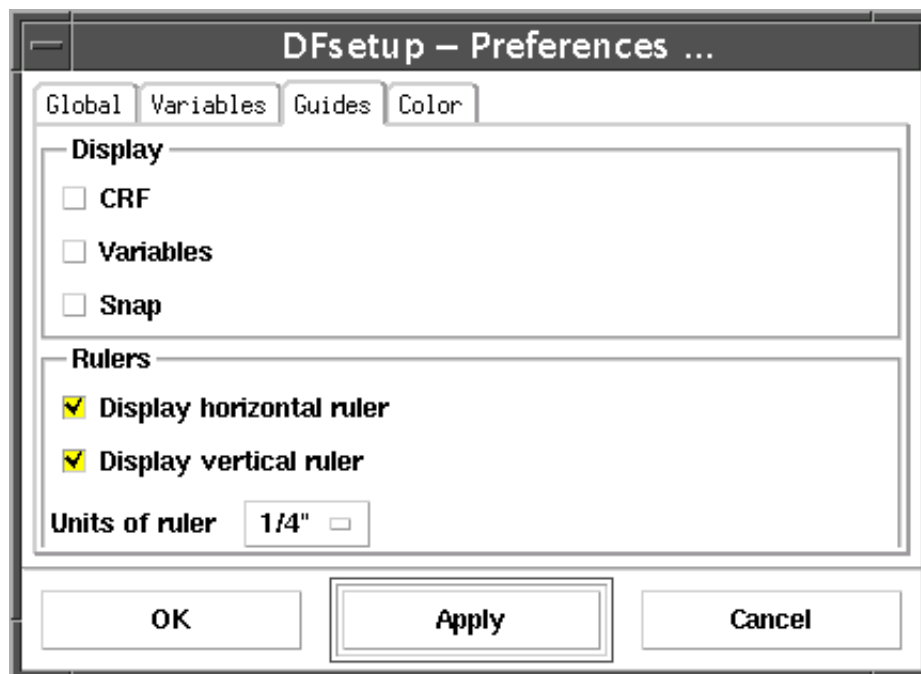
Figure 2.30. The horizontal ruler having 0.25 inch units of measure as displayed on the DFsetup screen.



The following steps should be taken to change the visibility of the rulers or change the unit of measure.

1. Locate Ruler within the Guides dialog.

Figure 2.31. The Ruler dialog box.



2. Change the visibility of the rulers.

Do this by toggling on or off the desired specifications for Display horizontal ruler, Display vertical ruler.

3. Set the units of measure.

Select the units (1/4", 1/10", 1mm) from the pull-down menu.

4. Click OK or Cancel.

Click OK to apply your changes to the current display or Cancel to discard them.

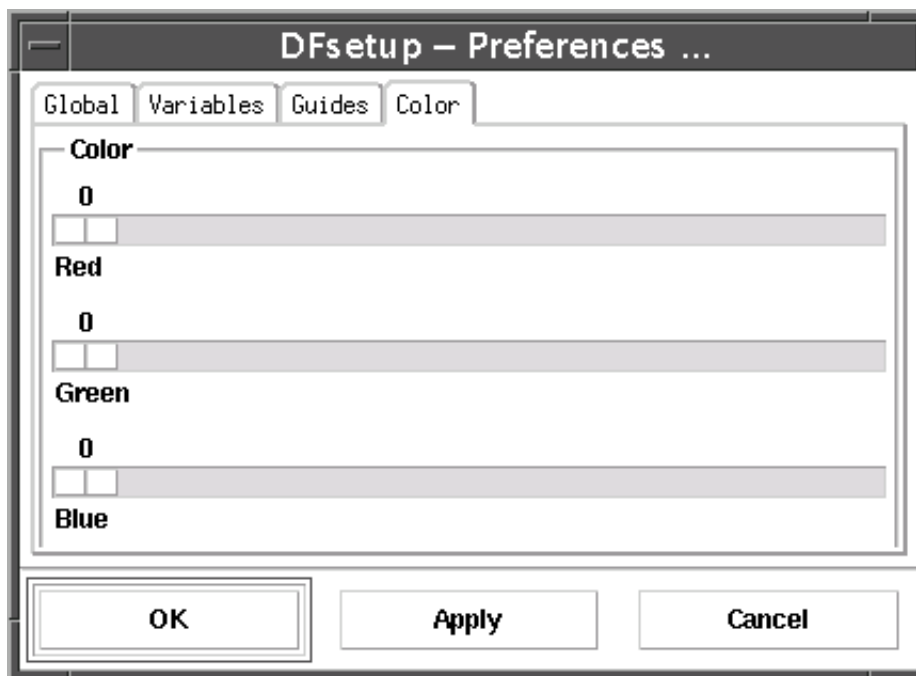
2.15.2.2. Color

You can specify the color of various features of the **DFsetup** window. These colors are in no way connected to the colors that can be specified in **DFvalidate**. You can change the color for the background, foreground, CRF, and variables.

To change the color of one or more features:

1. Choose the Color tab from the Preferences dialog.

Figure 2.32. The Color dialog box.



2. Select the feature for which you wish to change the color.

Choose Background, CRF, Variables or Foreground.

3. Use the left mousebutton to move the color sliders.

The color of each feature is determined by mixing intensities of the three primary colors red, green, and blue. Maximum intensity for each primary is 255 (full on); minimum intensity is 0 (full off). As you move the sliders in the Color dialog box, the color of the selected feature will also change in the main **DFsetup** window.

4. Select other features and change their colors as desired.

Do this by repeating steps 2 and 3 as necessary.

5. Click OK or Cancel.

Click OK to apply your color changes or Cancel to discard them. The color settings that you specify become part of your setup file the next time that it is saved. If you discard your changes, each of the feature colors is reset to the value it had before you invoked the Color dialog.

2.15.2.3. Identify the Variable on the Page

There are two methods for identifying the location of a new variable on the current page. The first, and more common, method is used when the response field on the CRF is constructed from one or more 0.25 inch squares. Using this method you click on each of the squares in order from left to right. The second method is used when there are no underlying data squares, typically for string or VAS variables. Using this method you drag out a bounding box that indicates both the location and the extent of the variable. For either method, it is not possible to position variable widgets outside of the visible page boundaries.

1. **Identifying Data Fields Comprised of One or More Squares.** The following steps are used to identify numeric, date, choice and check fields.

1. Position the cursor somewhere inside the leftmost square.

The cursor may appear as an arrow or a cross-hair.

2. Click the left mouse button.

When you click inside the square, setup looks outwards from that point for the 4 edges of the box. If the edges can be found, the corners of the square are marked with handle symbols.

3. Click on the remaining squares in the order that they define the data field.

It is important that you keep the order of the squares in mind. For numeric and date fields, they should be selected in straight left-to-right order. If you are defining a sequence of horizontal choices, again straight left-to-right order is appropriate. If you are defining a choice variable with squares laid out on the CRF in a single vertical column, select them from top-to-bottom. If the choices are laid out in a rectangular grid with two or more rows and columns, select the squares left-to-right within each row moving top-to-bottom, or top-to-bottom within each column, moving left-to-right. Use the ordering that makes the most sense for the field and try to maintain consistency when defining other variables that have the same rectangular grid choice pattern.

2. **Identifying a Bounding Box for String and VAS fields.** If a text field consists of a single rectangular box it can be selected by clicking anywhere inside the box. More commonly, text fields appear on the CRF with a thin horizontal line to indicate where the response should be printed, which is also the case for visual analog scales. To indicate the location of such fields perform the following steps.


1. Position the cursor over the leftmost end of the horizontal line.

Don't worry if the cursor is not positioned exactly on the line. Once the variable widget is completed, it can be 'snapped' into the correct position.

2. While holding down the left mousebutton, drag out a rectangle up and to the right.


You will notice that you cannot drag the height of the rectangle greater than 0.25 inch. This restriction is imposed by setup and matches the height of data boxes.

Figure 2.33. A bounding box is dragged the length of the string field.

If abnormal, comment 

3. When the right edge of the rectangle is approximately coincident with the right end of the line, release the left mousebutton.

Figure 2.34. Once the mousebutton is released, you will see a bounding box defining the size of the string field.

If abnormal, comment 

Although not visible, the bounding box contains four small handles that indicate the corners of the rectangle.

Figure 2.35. Handles defining the four corners of the rectangle.

If abnormal, comment 

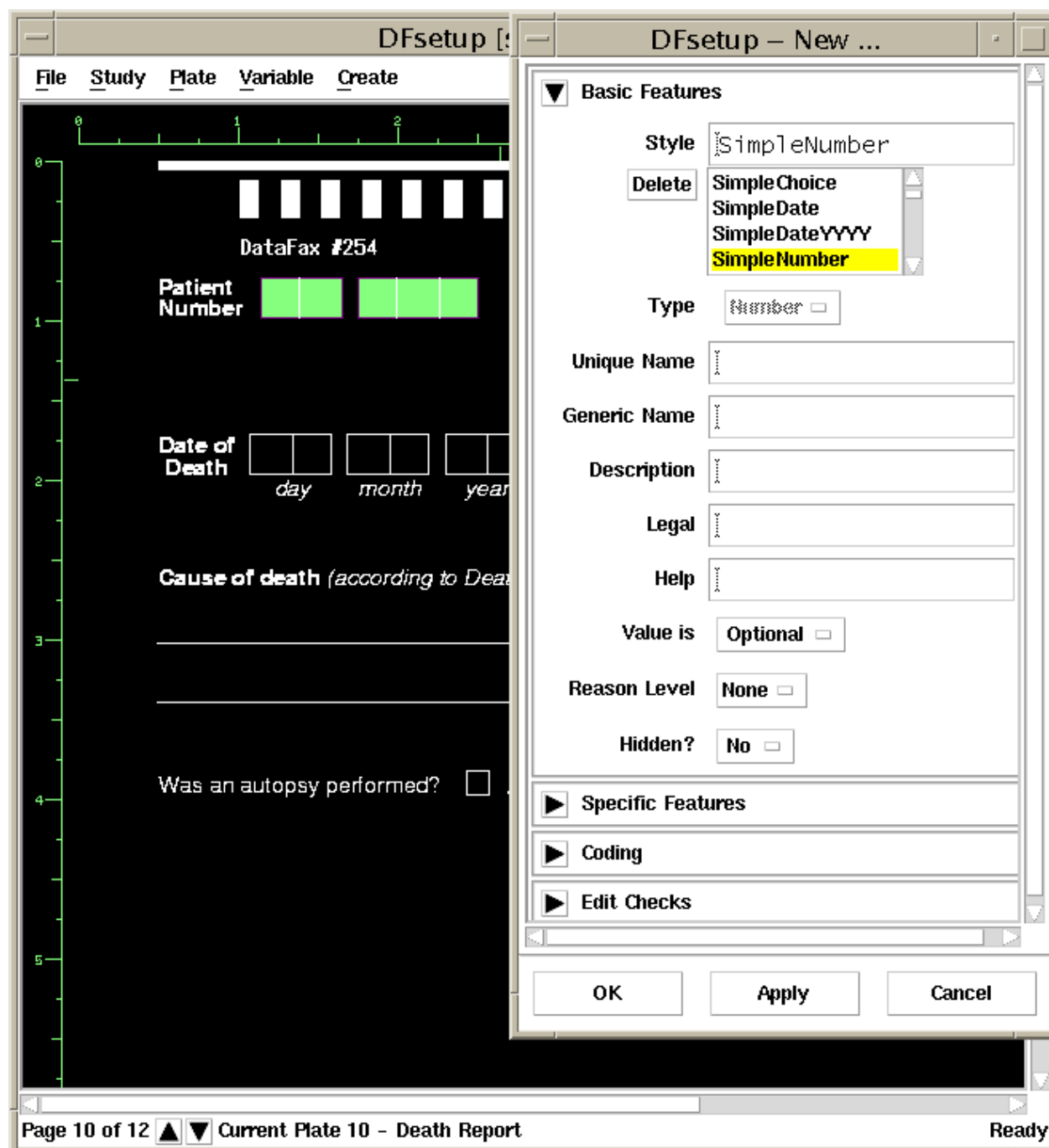
If you want to change the size or location of the rectangle, you can discard the bounding box by clicking the left mousebutton over a blank (background) area of the CRF. This will erase the box and you can begin again from step 1.

This method can also be used to locate a numeric field which is not comprised of individual boxes or to add a hidden variable which is to appear somewhere on the data entry screen, but does not appear on the CRF. This is described further in Creating Hidden Variables.

2.15.2.4. Define Variable Attributes

After identifying the location of the new variable on the page choose New from the Variable menu to bring up the New dialog box. This window has exactly the same appearance as the Style window you have already used to define variable styles. The example below shows a Patient Number data field which has been located (note the shading defining the Patient Number field) and the New dialog box which you need to complete.

Figure 2.36. Variable attributes are defined by completing the New dialog box.



DFsetup makes an initial guess at the data type of the identified data field and selects the corresponding **Simple** style as the initial style for the new variable. In making its guess, **DFsetup** uses the following rules:

- If a single 0.25 inch square is identified, the assumed data type is Check.
- If a single rectangle that is at least twice as long as it is high is identified, the assumed data type is String.
- If three groups of two 0.25 inch squares each are identified, the assumed data type is Date.

- If at least two adjacent 0.25 inch squares are identified, the assumed data type is Numeric.
- Otherwise, the assumed data type is Choice.

You can, and frequently will want to, change the style that **DFsetup** has initially suggested to some other style.

1. Select a Style

The definition of a new variable must always begin with a variable style from the style catalog. You then complete the definition by specifying attributes for the new variable which are not already defined in the style.

To select a style for your new variable:

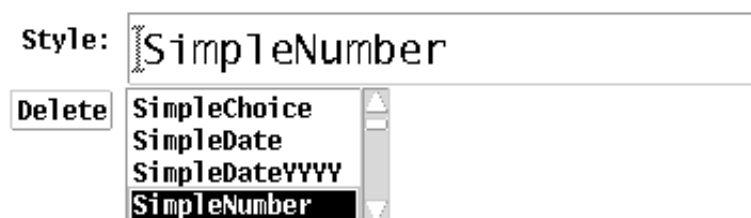
- Determine the underlying data type of the new variable.

Don't be fooled by the appearance of the data field on the CRF. For example, if a model number for a medical device is represented by a series of 0.25 inch boxes on the CRF, but the model number can contain alpha-numeric characters, this field should be defined as type String, not type Number.

- Choose a style having this data type from the style catalog scrolling list.

Search, by name, for the style that you want. In this example, the initial guess by **DFsetup** was numeric data type and hence the `SimpleNumber` style was chosen. But, having already defined a `PatientId` style for this particular numeric data type, you would override this initial guess by choosing the `PatientId` style from the Style Catalog menu.

Figure 2.37. The style chosen from the style catalog scrolling list appears as the name in the Style field.

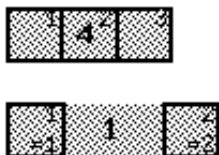


2. Define Features

The Basic attributes defined by the selected style will be displayed in the appropriate fields of the Basic Features window and then stippled over to indicate that their values cannot be changed here. All of the Basic Features not set in the selected style can be entered in the boxes provided. A variable name and description must be provided. The short variable name, legal values and help message are all optional. Edit Checks and Specific features are optional, and can be set for each variable as already described for creating variable styles in Creating a New Variable Style. When you have finished entering features, click OK or Apply to save your specifications or Cancel to discard what you have entered.

3. Apply the definition and examine the variable widget

When you click OK or Apply to end your definition of a new variable, a stippled pattern, referred to as a variable widget, will appear over the location of the variable on the CRF image, as shown in the following examples.

Figure 2.38. Variable widgets for a numeric and choice variable

2.15.2.5. The Variable Widget

The variable widget shows the complete extent of the variable's location and includes all parts of the variable. The larger number in the middle of the widget is the variable's ordinal position on the current page. The smaller number near the top-right corner of each square is that square's ordinal position within the variable. This ordering matches the order in which you selected the squares when indicating the location of the variable. For choice variables the bottom-right corner of each square contains the numeric code corresponding to that choice preceded by an equals '=' sign. It is a good idea to examine the widget after creating a new variable to confirm that it has been defined as you expected.

2.15.3. Moving Between Variables

2.15.3.1. The Current Variable

The 'current' variable is the variable that all edit commands, including delete, will be performed upon. The current variable is marked with handle symbols at its four corners. With the Variable feature of the Guides dialog turned off, (see Variables), the appearance of the current variable is illustrated in Figure 2.39.

Figure 2.39. Handle symbols indicate the current variable when Guides > Variable is turned off.

With Guides > Variable turned on, the same current variable has the appearance illustrated in Figure 2.40.

Figure 2.40. A stippled variable widget indicates the current variable when Guides > Variable is turned on.

You can make any variable defined on the current page the new current variable simply by clicking on it with the left mouse button. If you select a new current page, the current variable becomes undefined.

2.15.3.2. The Previous and Next Variables

You can also move sequentially through all defined variables on a page by using the Next and Previous commands from the Variable menu.

- **Choose Previous from the Variable menu to make the previous variable the current variable.** The previous variable is the one whose variable number is one less than the variable number of the current variable. If the current variable has number 1, the previous variable is the last variable on the current page. If you have Audible Bell enabled in your window environment, you will hear a bell when moving from the first variable to the last variable.
- **Choose Next from the Variable menu to make the next variable the current variable.** The next variable is the one whose variable number is one greater than the variable number of the current variable. If the current variable is the last variable on the page, the next variable has number 1. Again, if you have Audible Bell enabled in your window environment, you will hear a bell when moving from the last variable to the first variable.

2.15.3.3. Repositioning the Current Variable

You can reposition a variable simply by dragging its widget to a new location.

To reposition a variable:

1. Click the desired variable to make it the current variable.

Select the current variable using the left mouse button.

2. Grab the variable widget by holding down the left mousebutton.

When you grab the variable widget only its outline is displayed.

3. Drag the widget to its new location.

If you are positioning the variable over a specific location on the CRF image, you can use the Guides > Snap feature (see Snap) to aid you in positioning the variable.

4. Release the left mouse button.

Releasing the left mouse button will release the variable and fix its location.

2.15.4. Modifying a Variable

A variable definition can be modified at any time, including after the study has started, although the latter is not recommended. To modify an existing variable click on it to make it the current variable and then choose Edit from the Variable menu. The same dialog boxes used to define a new variable are edited to modify the definition of an existing variable. Remember to click Apply in each dialog box to save your changes and to select Save from the File menu before you exit from **DFsetup** to update the setup definition.



Warning

If your DataFax study is already in operation, you must recreate the study schema before the changes will take effect. But be careful if you already have data in the database for the modified variables. You can change variable attributes which do not effect the database with impunity, but modifications which alter the format, coding or length of a data field will only be applied to CRF pages received after the modifications were applied. Thus it is necessary to take your study off-line, and modify the existing database to conform to the changes before you apply them through the Create menu and before you allow any new CRFs to be processed under the modified ICR and Validation Tool screens. If you need to modify an existing study see Reordering Variables.

2.15.5. Deleting a Variable

Again, be careful when deleting variables for a study which is in progress. If the database contains records for the plate on which variables are being deleted the database will have to be restructured to coincide with the new layout of data fields (see Reordering Variables).

To delete a variable:

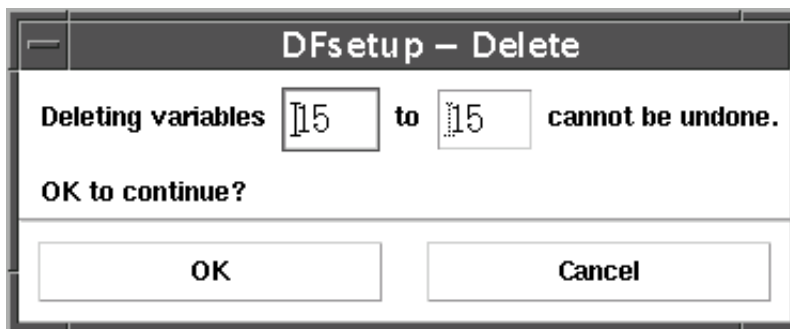
1. Select the variable.

Click on the variable to be deleted to make it the current variable.

2. Choose Delete from the Variable menu or press the Delete key.

The following pop-up window will appear.

Figure 2.41. A confirmation dialog box appears once you choose Delete.



By default, the variable number of the current variable is inserted into the text fields. Change the minimum and maximum variable numbers if you want to delete additional variables as well.

3. Click OK to delete the variable(s) or Cancel to keep it.

When you delete a variable, the remaining variables are automatically renumbered. Normally the variable which followed the deleted variable becomes the new current variable. However, if you delete the last variable on the page, the variable which preceded it (i.e., the second last variable) becomes the new current variable. If you delete the only variable on the page, the current variable becomes undefined.

2.15.6. Copying and Pasting Variable Definition(s)

Sometimes you will need to define several variables which have nearly the same definition, perhaps differing only in a few characters of the variable Unique name and description. For example, a list of patient exclusion criteria might be named ex1, ex2, ex3, etc. and have descriptions Exclusion #1:, Exclusion #2:, Exclusion #3:, etc. After defining the first variable of such a set the easiest way to create the others is to copy the first variable, paste it over each of the remaining data fields to be defined and then edit each of the pasted variable definitions. You may also have a line of several variables defined on one page that needs to be replicated on another page. After defining the first group of variables, the entire group can be copied from one page and then pasted on the other page.

To copy and paste one or more variables:

1. Select the variable(s) to be copied.

If you want to copy a single variable, make sure that the variable to be copied is the current variable. If it is not already the current variable, make it so by clicking on it.

If you want to copy a group of variables, hold down the Shift key and the left mouse button while you drag a bounding rectangle around the variables to be copied.

If you want to copy all of the variables on the current page, choose Select All from the Variable menu.

The selected variables are indicated by the handles on the corners of their widgets.

2. Choose Copy from the Variable menu.

The currently selected variables are copied into the copy buffer, replacing any selected variables which might have been copied there previously.

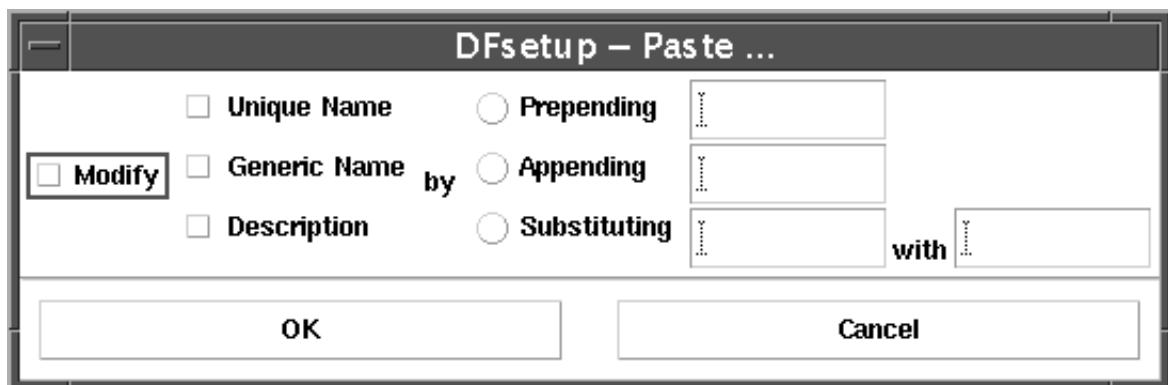
3. Go to the destination page.

If the page that you want to copy the selected variables to is not the current page, make it the current page.

4. Choose Paste from the Variable menu.

The following dialog box is displayed:

Figure 2.42. The Paste dialog box invoked from the Variable menu.



Most often when you copy variables you will need to edit the new variable definitions to change the `Unique name` and other features. This dialog allows you to specify simple edits that can be applied to each of the variables as it is copied. You can edit the variable `Unique name`, `Generic name`, and/or `Description`. The edits you can perform include prepending or appending a character string or substituting part of a character string with another. However, only one of the three edits can be performed at a time.

- Specify the edits to be applied and click OK to paste the new variables, or click Cancel to cancel the paste operation.

If you do not wish to apply edits at this point, make sure that the `Modify` box is not checked.

To indicate edits, first select which features you wish to apply the edits to. You can edit the `Unique name`, `Generic name`, and `Description` features. Next indicate the type of edit by supplying the arguments in the appropriate text fields. Finally click OK to apply the edits and paste the new variables.

If for example, you had copied a variable named `ex1` having a description `Exclusion #1`, you could specify:

Figure 2.43. Specifications for character substitution during a variable Paste operation.

The screenshot shows a dialog box titled "DFsetup - Paste ...". It contains three main sections for editing variable attributes: "Unique Name", "Generic Name", and "Description". Each section has a checkbox to enable editing and a radio button to select the type of edit (Prepending, Appending, or Substituting). In the "Unique Name" section, the "Unique Name" checkbox is checked and "Prepending" is selected. In the "Generic Name" section, the "Generic Name" checkbox is unchecked and "Appending" is selected. In the "Description" section, the "Description" checkbox is checked and "Substituting" is selected. The "Substituting" section has two text input fields: the first contains "1" and the second contains "1", with the word "with" between them. At the bottom of the dialog are "OK" and "Cancel" buttons.

to paste a new variable having name `ex2` and description `Exclusion #2`.



Warning

When substituting a new character string for an old one, only the first occurrence of the old string in each attribute field, is replaced by the new string. Also, if the edit you are applying results in the variable names or description being longer than their defined limits, DataFax will apply the modification, truncate the result at the limit and display a warning message that truncation has occurred. You may then wish to go back and manually edit the truncated variable attribute.

- Position each pasted variable.

When you copy and paste variables within the current page, the new pasted variables are positioned slightly below and to the right of the copied variables. On the other hand if you copy variables across pages, the pasted variables are positioned in the same location on the current page as the copied variables were positioned on the page from which they were copied.

If necessary, re-position pasted variables following the directions given in Repositioning the Current Variable.

7. Edit each pasted variable definition.

If necessary, edit each pasted variable by making it the current variable and then choosing Edit from the Variable menu.



Note

It is possible to change the variable description length through the DFsetup - Preferences dialog at any time during study setup. It is therefore possible that performing the copy/paste operation following a reduction to variable description length (i.e., from 40 to 25 characters), will lead to overlength variable descriptions in the database. If this occurs, you will be warned of overlength descriptions by the appearance of the Overlength Variable Descriptions dialog when you perform the paste operation (see Overlength Variable Descriptions).

2.15.7. Importing Pages and Styles from Another Study Setup

If you have already defined one or more study setups for other DataFax studies, you can save yourself considerable time and effort in defining your new study by re-using page and/or style definitions from the completed setups. Just as you have probably re-used page templates in your desktop publishing package to produce new paper CRFs, you can re-use previous DataFax setup definitions.

DFsetup allows you to import styles and pages into your current study setup from other study setups. When a page definition is imported from another study, all of the variable styles referenced on that page are automatically included for import. Imported styles replace existing styles in the current study that match by name if the Replace existing styles with imported styles of same name check box is checked. Other imported styles are simply added to the style catalog for the current study.

Each imported page definition includes all of the variables defined on that page. If the plate number of the imported page does not already exist in the current study, a new page definition must be added. If the plate number is already defined in the current study, the complete page definition is replaced if the Replace existing plates with imported plates of same number check box is checked. The existing background image for the page(s) will not be replaced, however.



Note

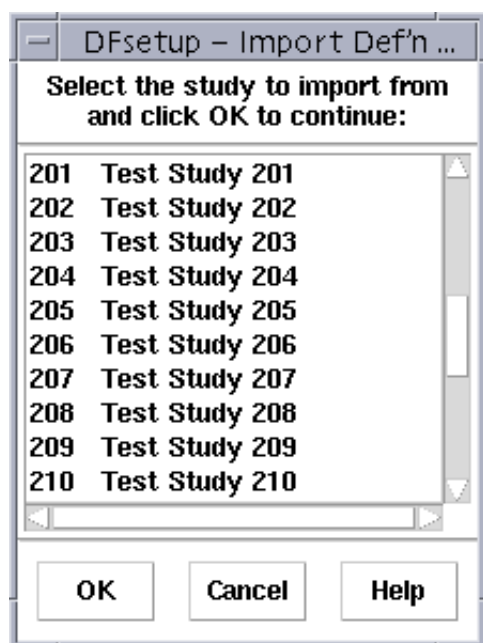
When importing plate definitions from another study, DataFax assumes the plate numbering of the original plate and the new plate will remain the same. As a result, the import default setting is to have the Replace existing styles... and Replace existing plates... check boxes turned on. If you do not wish to replace existing plates and styles, be sure to de-select one or both of these options.

To import pages and styles from another study setup:

1. Choose Import Def'n from the Study menu.

A scrolling list of currently defined studies, except of course the one that you are defining, is displayed in a dialog box.

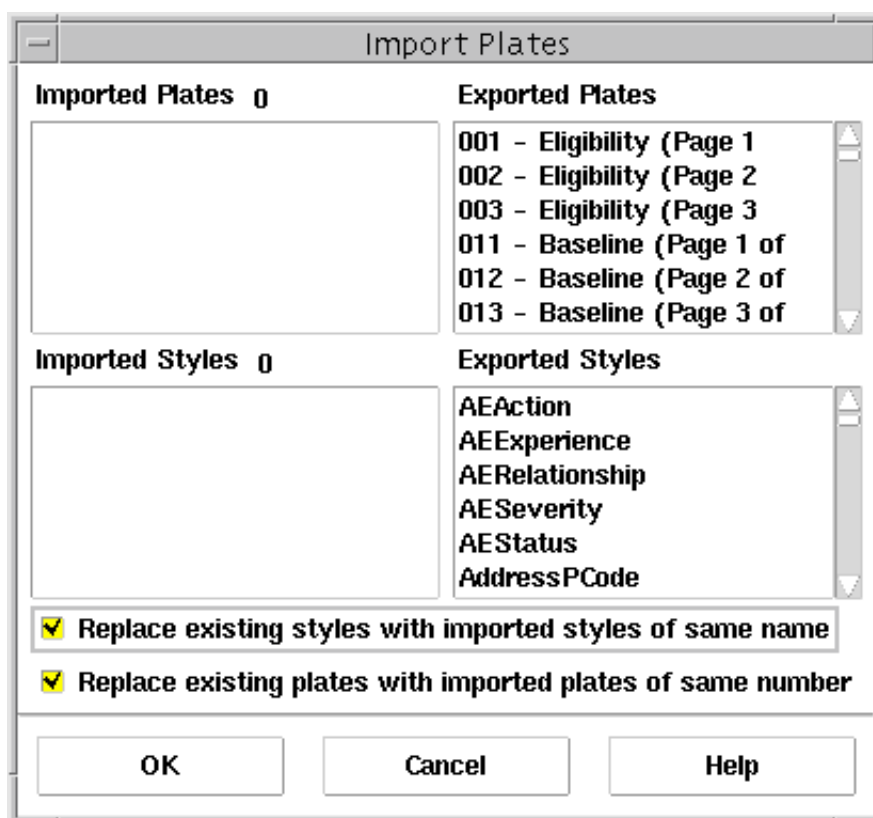
Figure 2.44. The Import Def'n dialog box will first display a list of defined DataFax studies.



2. Choose the study from which you wish to import setup definitions.

Click on the study description in the scrolling window (as illustrated above) to select the study from which you wish to import setup definitions. Then click OK. Alternatively, you can just double click on the description in the scrolling window. **DFsetup** will attempt to load the setup definition file for that study. If the load operation succeeds, you will be presented with the following dialog box:

Figure 2.45. The Import Plates dialog box provides a list of plates and variable styles defined for the chosen study.



3. Move plates and/or styles into the Imported Styles and/or Imported Plates scroll lists.

To move a plate or style, select it and click an arrow, or double-click the plate or style.

When you import a plate, all of the styles referenced on that plate are automatically added to the Imported Styles scrolling list.

4. Toggle on or toggle off the Replace existing styles with imported styles of same name box and the Replace existing plates with imported plates of same number box.

By default, where an existing style and an imported style have the same name, the imported style overwrites the existing style. Where existing plates and imported plates have the same number, the imported plate overwrites the existing plate's definitions. Toggle the box off to override this behavior and discard imported styles/plates that have the same name/number as an existing style/plate.

5. Click OK.

The imported plates and styles are retrieved from the study and added to the current study setup.

2.15.8. Creating Hidden Variables

Occasionally you may want to create a data entry field which does not appear on the study CRFs. These are variables which will exist in the database but which are hidden to anyone simply examining the study CRF. For example, you might want to include a numeric data entry field to code text fields (e.g. symptoms, drug names, comment fields) but not want to clutter the CRF with data fields which will not be completed by the clinical investigators. Another common example is the use of a check field to indicate that the investigator has signed a CRF page. This might appear on the CRF as shown below.

Figure 2.46. Data fields for collecting Investigator sign-off information, as they appear on the CRF.

investigator's signature. _____ Date.

In this example, the CRF page includes a line for the investigator to sign. We may want to record in the database that the page was signed but feel that it would be too time consuming and unnecessary to have the data entry clerk attempt to read the investigator's signature and type it for all pages that require a signature. Instead we could create a single box for a check field over the space reserved for the signature and simply have the data entry clerk check this box if the page was signed.

To create a hidden variable:

1. Drag out a rectangle defining the new variable's location and extent.

To create a check box hold the left mousebutton down at the desired location on the CRF image. Then drag out a rectangle about 0.25 inch square. Release the mouse button to complete the rectangle. If you want to erase it and start over just click left anywhere there is empty space on the CRF image.

2. Define the new variable.

Choose New from the Variable menu to bring up the **DFsetup** New dialog box, and proceed to define the new check variable as described in Define Variable Attributes. When completed, the setup window should look like the example shown below.

Figure 2.47. Investigator signature field defined as a Check data type in DFsetup.

investigator's signature. ☐ _____ Date.

You should recognize this as identical to the description of how a text field is created (see Text). You can create any of the 6 data types in this way. To create a numeric field drag out a rectangle long enough to hold the number of digits that it will contain, allowing 0.25 inch for each digit. For a choice field drag out a 0.25 inch square box for each of the desired response options. For a date field formatted as yy/mm/dd drag out a rectangle either 1.5 inches long, if the date format does not include delimiters, or 2 inches long if it does. For a visual analog scale (VAS) drag out a rectangle 4 inches (10 cm) long.

2.15.9. Reordering Variables

Variables can be defined on each CRF plate in any order. But the final order, with which the data entry screens are created, must correspond to the order in which the data fields on each plate are to be completed during data entry. If you define variables in this order to start with then all is well, but if not you will need to reorder the variables on the page.



Warning

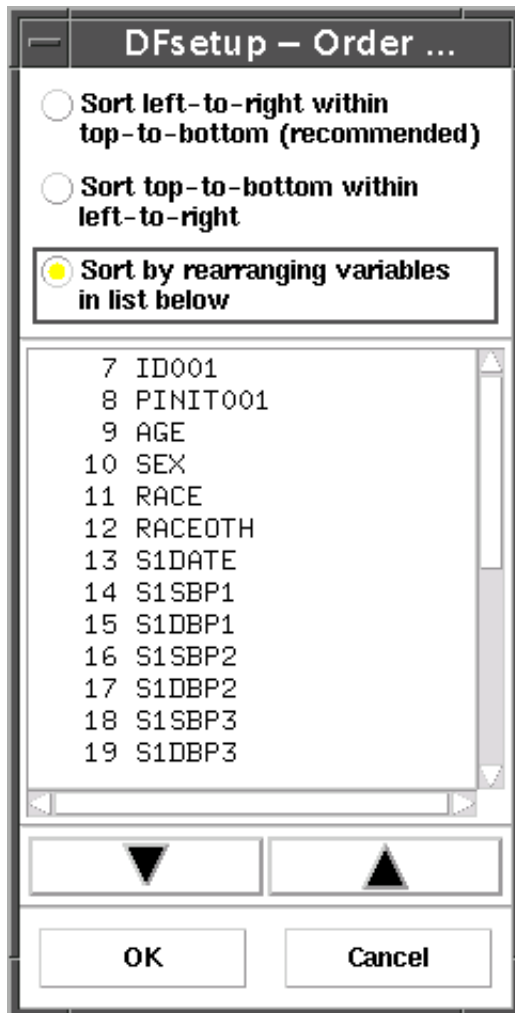
If you reorder the variables on a page after the study has started and there is data for the effected plate in the database, then the existing data records will have to be reformatted to conform to the new variable order before you modify the study setup. See Modifying the Setup of a Study in Progress for more information.

To reorder the variables on the current page:

1. Choose Order from the Variable menu.

The following dialog will appear:

Figure 2.48. The Order dialog box used to assign variable order within a CRF page.



2. Select the ordering method.

The default ordering method is Sort by rearranging variables in list below which allows you to re-order variables by re-positioning them in the list. To re-order a variable in this way follow these additional steps.

- a) Select the variable in the list.

Variables are listed in ascending order by number and `Unique name`.

- b) Use the up-arrow or down-arrow buttons to move the selected variable to a new position in the list.

As you move the variable, its number changes.

Repeat these steps for each variable that needs to be re-ordered. This method of variable ordering will only allow you to reorder one variable at a time.

The ordering method Sort left-to-right within top-to-bottom orders the variables left to right across the page as we move down a page. Choose this method to quickly re-order all variables on the current page in this fashion. You can subsequently switch back to Sort by rearranging variables in list below to make further adjustments.

The ordering method Sort top-to-bottom within left-to-right is beneficial for forms in which questions are presented column-wise and the order of data entry should proceed down each column before beginning the next column.

3. Click OK or Cancel.

When you have selected all of the variables on the page click OK to save the new ordering, or Cancel to discard it.

2.15.10. Modifying the Setup of a Study in Progress

This section discusses the issues to consider and steps to follow when modifying a DataFax study in progress, i.e. a study for which CRF pages and data records have already been received. It is important to understand the impact that changes will have and the effort required to completely implement them before proceeding to make changes in **DFsetup**. The key issues to consider include the following:

- **Integrity of the Study Database.** The data in a DataFax database is comprised of data records each containing an ordered list of data fields, which match the order of the fields defined in **DFsetup**. Thus adding, deleting, or re-ordering data fields will result in an inconsistency between data collected before and after the change. For this reason it is recommended that data fields not be added anywhere except at the end of existing data records, and that existing data fields not be deleted or re-ordered.

If you decide to ignore this recommendation you will need to reformat the existing data records to match the new ordered list of data fields defined in **DFsetup**. DataFax does not do this automatically. The task is made even more involved by meta-data records, i.e. QC notes (plate 511) and reasons for data change (plate 510). Meta-data records contain a field which identifies the data field on the CRF plate to which that meta-data record is linked. To maintain this link, any meta-data records effected by a change in the ordering of data fields in **DFsetup** must be identified and modified to hold the new data field number.

Database inconsistencies can also arise if the coding, format or length of a data field is changed. Increasing the length of a data field will not create an inconsistency but reducing the storage length will be a problem if values exceeding the new length exist in the database. DataFax will not automatically recode or reformat existing data fields if you redefine these attributes. Thus to keep data collected before and after the change consistent, the effected data records must be exported, modified to apply the changes, and then re-imported to replace the existing data records. Again, such changes are not recommended.

- **Audit Trail Reports.** If the study is being conducted under FDA or other regulatory requirements, auditors may want to review audit trail reports of database changes. The DataFax journal files consist of copies of the data records as they existed when they were written to the study database. If field specifications are changed, the specifications written to the study schema may no longer be accurate for data records written to the journals before the change was made. For example, if field number 20 was deleted after a certain date, all fields from that position on will have a different meaning in data records journaled before and after the change; and if fields are recoded, the meaning of the codes

written to the journal files will differ before and after the change. DataFax does not keep a record of what the specifications were for the fields on the plate before you made the changes. When the audit trail report program is run it will apply the current field definitions to all data records in the journal files, including the those created before, as well as after, the change. The result will be an incorrect audit trail report for any case in which field position or coding has changed.

Deleting plates from the setup will also have negative consequences for audit trail reports, as the audit trail program will no longer be able to find the field definitions for these plates in the journal files. The resulting output will identify changes in data values without identifying which data fields were changed.

If this is an issue, or if you want to preserve accurate audit trail reports for your own purposes, do not delete, insert, reorder or recode data fields on any plate, and do not delete plates from the setup even if they are no longer being used. It is safe to add data fields to the end of a plate as this has no impact on the order of fields in the data records. But this is the only field list modification you should make.

- **Legal Values.** If the legal value list for a data field is changed to be more restrictive it is possible that some existing clean data records may now have illegal values. They will not be changed to dirty unless they are revalidated and committed to the database.

Similarly, if the legal value list for a data field is changed to be more liberal it is possible that some existing dirty data records may no longer have illegal values. Again, they will not be changed to clean unless they are revalidated and committed to the database.

- **Edit Checks.** Edit checks can reference data fields by field number or by name. Changing either of these will thus require an evaluation of the impact this may have on existing edit checks.
- **QC Reports.** When a QC note is created the current data field description is stored in a field in the QC note. These descriptions are used in QC reports to identify the field to which the query is related. Data field descriptions can be changed if needed to improve the identification of the data field, but such changes will only appear on QC notes created after the change was made.

One negative consequence of changing the data field description is that this will result in different descriptions for the same data field for QC notes created before and after the change.

2.15.10.1. Restructuring a Study Database

If after considering the issues described above you decide to add, delete or reorder the data fields on one or more plates, or to modify the coding, format or storage length of one or more data fields, you will need to make the necessary modifications to the corresponding data records that currently exist in the study database. This must be done by your DataFax system administrator or by a programmer. Even if the planned modifications do not require that existing data records be reformatted, you should have all users stop working on the study and exit from their study DataFax tool box before you proceed to modify **DFsetup**. The changes which you make that affect the appearance or operation of data screens in **DFvalidate** will not appear until a user exits from an existing **DFvalidate** and then restarts it.

To modify a study which is in progress you and your DataFax system administrator must proceed as follows:

1. Plan the change over date.

If you are revising existing CRF plates and not just adding new ones you will need to plan a change over date. All clinical centers must be notified of the date on which they are to stop using the old forms and start using the new ones. If the change over is carefully planned and any programming needed to reformat data files is performed and tested prior to the change over date, it should be possible to accomplish most revisions to the database and study setup in one hour or less.

2. Clearly specify the planned modifications.

You will have to describe to your system administrator or programmer exactly how the setup will be modified. For each CRF plate on which changes will be made, identify the following:

- **Which variables, if any, will be recoded.** If you have decided to modify the coding of choice fields then some cases may need to be recoded. If you are simply adding additional options with new response codes then the existing database will not be affected, but if you have decided to switch the coding of no/yes, from 1/2 to 2/1 for example, then any existing data records must be modified to conform to this change.
- **Which variables, if any, will be deleted.** The data fields corresponding to deleted variables must be removed from existing data records.
- **Which variables, if any, will be added.** If new data fields are to be added you will need to specify where they will be inserted and whether they should be left blank or entered with some value. If for example, you are adding a hidden variable so that a comment field can be tagged with a hidden numeric field for coding purposes, then it makes sense to leave the new field blank in the existing data records. After the setup has been modified and the study is re-enabled, someone can then review all of the existing records for that plate and enter the desired codes. If on the other hand you are adding a field to capture new information on future patients which you do not intend to try to recover for existing patients then you may want to fill this field with some code to represent old patients, or with the DataFax Not Available (*).
- **Which existing plates, if any, will be deleted.** If an entire plate is deleted you should also delete the corresponding data and index files from the study data directory, or at least move them to some back-up location.
- **Which variables, if any, will be reordered.** The order in which variables are entered in Validation Tool data screens corresponds to the order in which they are stored in data records. Thus any change in the ordering of variables on the page will require a restructuring of any existing data records.
- **Which variables, if any, will be reformatted.** Variables may need to be reformatted, e.g. to change a date format, or reduced in length, e.g. truncated to fit a shorter storage specification.

3. Obtain exclusive access to the study.

The changes to the setup and database need to be made in an environment where no new CRFs are arriving and no other users are performing data review or data entry.

Exclusive access to the study is obtained by issuing:

```
% DFaccess.rpc -x -s study#
```

at the command-line, and then confirming that exclusive access has been obtained:

```
% DFaccess.rpc -q -s study#
Study study# is in exclusive access mode.
```

With exclusive access, the current user is assured that they can make the needed changes without interference.

4. Export data records from each of the affected plates.

The existing data records must be exported from the database using **DFexport.rpc**, as described in *DataFax Programmer Guide*, *DFexport.rpc*. Do not attempt to edit the database directly. This will destroy the link between the edited data file and its corresponding binary index file. This will corrupt the database for the edited file.

If new fields are being added to or deleted from an existing plate, **DFexport.rpc** can be used to pre-format the exported records so that their content matches the planned changes. For example, if a new variable is being inserted between fields 30 and 31 on plate 2 of study 254 and that variable will receive a default value of * for all existing records, the following **DFexport.rpc** command might be issued:

```
% DFexport.rpc -s "primary secondary" -p -f "1-30,'*',31-NF" 254 2 plt002.new
```

The output file, `plt002.new`, created by **DFexport.rpc** in this example, is correctly formatted for re-import to the study database following the modification of the study setup to include the new variable.

5. Reformat the exported data records for each of the affected plates.

If it was not possible to pre-format the data records using the capabilities offered by **DFexport.rpc** as described in the previous step, it will now be necessary to reformat them.

The exported data records can usually be reformatted quite quickly and easily using `awk`. This is a simple programming language which was designed to operate on field delimited data records like those which comprise a DataFax study database. The DataFax **DFget** (*DataFax Programmer Guide*, *DFget*) program can also be used for field and record manipulation.

6. Export the meta-data records

The existing QC notes (plate 511) and reasons for data change (plate 510) must be exported from the database using **DFexport.rpc**, as described in *DataFax Programmer Guide*, *DFexport.rpc*. Do not attempt to edit the database directly. This will destroy the link between the edited data file and its corresponding binary index file. This will corrupt the database for the edited file.

The following **DFexport.rpc** command will export all QC notes:

```
% DFexport.rpc -s "all" 254 511 plt511.new
```

and the following **DFexport.rpc** command will export all reasons for data change:

```
% DFexport.rpc -s "all" 254 510 plt510.new
```

7. Reformat the meta-data records.

The plate and field numbers that link each QC note and reason for data change to a specific database field, are stored in fields 5 and 8 respectively, of the meta-data records. If data fields have been inserted, deleted or reordered, some of the field numbers in the meta-data records will need to be changed to maintain the link between the meta-data records and the data fields to which they refer.

This can be done using `awk` or some other programming language. The program needs to identify those meta-records that are effected, update the field number in field 8, and write the modified records to a file. This can be tricky if multiple changes were made, so make sure you have an accurate map of the changes required, and that you verify that the correct links have been preserved in the output you create for this step. Also make sure that field 8 is the only field you change. The modified meta-data records must retain their structure and contents so that they can be imported to the database to replace the old version.

8. Modify the study setup.

Next, modify the study setup using **DFsetup**. If one or more pages of the study CRFs are modified to add or delete variables, or to reformat the existing layout, you must import the modified pages and add, delete and reposition variables as needed in **DFsetup**. Importing the revised pages is accomplished as previously described (see Importing CRFs). Remember to check the box in front of Overwrite existing pages to overwrite the old images of the modified pages with the new versions. After importing the revised pages move to them and perform the necessary reformatting.

All of the variables previously defined for the old version of the page will appear over the new revised background image, but in the same location they had on the old version. To reposition a variable click on the variable widget to select it and then while holding down the left mousebutton drag the widget to the new location. Setting Snap from the Guides dialog will make this easier (see Snap).

If you have revised a CRF plate by adding and/or deleting data fields, you will need to define new variables for the added data fields (as described in Defining Variables) and/or delete variables which have been removed from the CRF (as described in Deleting a Variable). If you delete variables or add new ones it will probably be necessary to reorder the variables on the page after all deletions and additions have been performed (see Reordering Variables).

When you finish and exit from **DFsetup** it will automatically recreate the ICR template and other setup files to implement the changes you have made.

9. Import the reformatted data records.

Once a file of reformatted data records has been created, the database must be updated to include those records. This is done using **DFimport.rpc** (*DataFax Programmer Guide*, *DFimport.rpc*). All of the existing database records with matching identifiers must be replaced by their corresponding reformatted data records, and as such the appropriate option to use with **DFimport.rpc** is **-r**, as in:

```
% DFimport.rpc -r -v study# file_of_reformatted_records
```

and completing the example started above:

```
% DFimport.rpc -r -v 254 plt002.new
```

DFimport.rpc will report the number of records that were successfully replaced. If any records could not be replaced, they should be investigated using **DFvalidate** and re-validated manually.

10. Import the reformatted meta-data records.

After importing all of the reformatted data records, import the meta-data records that you have modified to preserve their reference to the correct data fields. Again, this is done using **DFimport.rpc** with the **-r** option to replace the existing meta-data records.

11. Test the modified setup.

Open **DFvalidate** to check your modifications. Retrieve all or a sample of records for each plate that you have modified and review cases until you are convinced that the modifications have been performed correctly.

12. Restore read-write access to the database

When all of the above steps have been completed, read-write access to the database can be restored and data management resumed.

Read-write access to the study is restored by again issuing:

```
% DFaccess.rpc -x -s study#
```

at the command-line, and then confirming that read-write access has been restored:

```
% DFaccess.rpc -q -s study#
Study study# is in read/write mode.
```

2.15.10.2. Adding New CRF Plates

A new plate can be added to a study at any time. This has no impact on the existing database because each plate is stored in a separate file. New plates are added to the study setup by importing the CRF image, editing the page definition for the new plate and then defining the variables on the plate. This is performed exactly as described for the initial study setup. As soon as the new plate is defined and the setup file has been saved DataFax will be ready to receive these pages and store them in the study database.

2.15.10.3. Deleting CRF Plates

It is not recommended that CRF plates which already contain study data be deleted. However, if after considering the issues described at the beginning of this section you decide to delete a plate this can be done by selecting Delete from the **DFsetup** Plate menu. If you currently have data in the database for this plate it will not be deleted. Only the definition of the deleted plate will be lost. As a result DataFax will not be able to process any further copies of the deleted plate and it will not be able to access those which currently reside in the database. You might thus want to delete or move the deleted plates from the database as well. This is a job for your system administrator or a trusted programmer. It would obviously be a disaster to remove the wrong data files, especially if they had been modified since the last backup was made.

2.15.10.4. Modifying Variable Definitions

Changes to variable definitions which do not have an impact on the existing study database, e.g. adding new edit checks, changing legal ranges, help messages, etc. can be made at any time. The changes will become active as soon as the setup is saved in **DFsetup**. Select the variable to be modified by clicking on it and then select Edit from the Variable menu. The Edit window which will appear is identical to the New window previously used to define the variable. Make the desired modifications to the Basic Features, Specific Features and/or Edit Checks windows and then click OK or Apply to save your changes.

2.15.11. Create Menu

DFsetup is used to create all of the study definition files DataFax requires before you can run a new study. You can review which files will be updated automatically by **DFsetup** by choosing Properties from the File menu. If you prefer, you can explicitly create any of the three files by selecting them from the Create menu. However, you do not need to explicitly create the > File Map, ICR Template, and Schema as **DFsetup** will update them as necessary from the setup definition you have created.

Other study definition files, (i.e., Center DB, Page Map, Visit Map) require information that is additional to the CRF setup definitions you have already specified. The editor dialogs that allow you to input the required information and create these files can be displayed by selecting Configuration from the Study menu, or by opening the **DFconfig** tool from the config itself. These setup files are stored in the study `lib` directory and may also be created or modified using a UNIX editor (e.g. vi) instead of the **DFsetup** editors if at some point you find this more convenient. For more information about the **DFconfig** editor tool, see

DFconfig.

2.15.12. Meta-Words

The following meta-words can be used within **DFsetup** to simplify the specification of variable attributes. They are particularly useful in the creation of variable styles.

DataFax Meta-Words

`$(legal)` `$(legal)` is evaluated to the list of legal values, defined by the value of the Legal field, for the current variable. It is most commonly used within help messages as shown in the following example.

Figure 2.49. A help message consisting of a variable's legal values as specified by the meta-word `$(legal)`.

Help: Legal values are: `$(legal)`

`$(ids)` `$(ids)` is evaluated to the concatenated list of patient id ranges from all Clinical Monitor centers defined in the centers database. This meta-word is primarily useful in the definition of legal values for the patient id field and is interpreted only within the context of the variable's legal range definition.

Figure 2.50. A legal range for the patient id field is defined as a field attribute by the meta-word `$(ids)` specification.

Legal `$(ids)`

`$(field)` `$(field)` is evaluated to the ordinal field number of the current variable within the current CRF plate. Numbering begins at 6 or 7 with the first variable which is not bar coded. The first variable will be either the sequence number (position 6) or patient ID number (position 7).

`$(plate)` `$(plate)` is evaluated to the CRF plate number on which the variable appears. It will always be 3 digits long and zero padded. For example, `$(plate)` for CRF plate numbers 1, 55 and 101 will be 001, 055, 101.

`$(rplate)` `$(rplate)` is evaluated to the CRF plate number on which the variable appears. It is the same value as `$(plate)` except that the value is not leading zero padded.

`$(workdir)` `$(workdir)` is evaluated to the study work directory.

`$(study)` `$(study)` is evaluated to the DataFax study number, which is always the first bar coded field on all study CRF plates.

`$(blank)` `$(blank)` is evaluated to those values which represent that the data field is empty. For numbers, dates and text fields `$(blank)` means that no numbers or characters have been entered into the field. For choice and check fields the absence of any choice or check are both coded as zero in the database and thus `$(blank)` is evaluated to zero. This meta-word can be useful when defining skip patterns as discussed in Skipping Irrelevant Variables During Data Entry.

<code>\$(choices)</code>	<code>\$(choices)</code> is evaluated to the numeric value codes for all of the possible response options for a choice variable, including the no choice code.
<code>today</code>	<code>today</code> is evaluated to the current calendar date from the computer's system clock. This meta-word is useful when defining a legal value range for a date field. In the following example the legal range specifies that the patient assessment date entered on the study CRF must lie somewhere between the beginning of the study on June 1st, 1990 and the date on which this page of the CRF is received by DataFax.

Figure 2.51. A legal range specification for a date using the meta-word `today`.

Legal:

2.15.13. Backups and Auto Saves

When you open **DFsetup** a backup of the existing setup file is created. Each time you select Save from the File menu the backup file is rewritten with the current version of your study setup.

In addition, the setup program makes a separate automatic save of your study setup at regular intervals while **DFsetup** is open. This is useful for recovering your work in the event of an unexpected hardware failure or power outage.

The default time interval between the automatic saves is 5 minutes. The first auto save is made 5 minutes after you make the first change to your previously saved study setup. Thereafter, at 5 minute intervals, the auto save file is overwritten with your current setup.

To enable/disable the automatic saves and/or change the save interval:

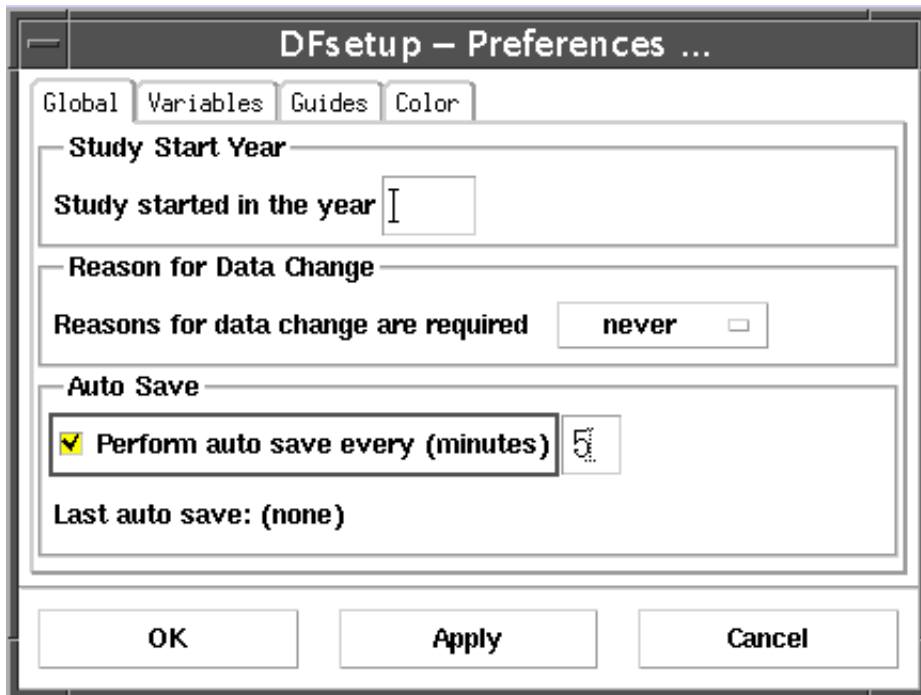
1. Open the **DFsetup** Preferences dialog.

Do this by selecting Preferences from the File menu.

2. Locate the Auto Save preference.

Select the Global tab in the Preferences dialog and locate the Auto Save section.

Figure 2.52. DFsetup's Auto Save dialog box.



3. Turn Auto Save on or off.

If the box in front of Perform auto save every (minutes) is checked, automatic saves will be made according to the frequency specified. If it is not checked, automatic saves will not be made. Clicking on this box toggles it on and off.

4. Specify auto save frequency.

To change the auto save interval, specify the number of minutes between saves. The minimum interval between saves is 1 minute; and the maximum interval is 99 minutes.

5. Click OK or Cancel.

Click OK to apply your changes or Cancel to discard them. If you have changed the auto save interval and auto saves are enabled, the auto save timer is reset to zero seconds, and the next auto save will occur after the specified interval.

2.15.14. Restoring Setup from the Backup

You can return to the backup file of your study setup by choosing Use Backup from the File menu. If you have not performed a save during the current session, you will be returned to the setup as it existed when you opened **DFsetup**. If you have performed a save you will be returned to the last saved version of the study setup.

2.15.15. Restoring Setup After a Program Failure

It is possible that on rare occasions you will experience unexpected program errors during your use of **DFsetup**. It is also possible that other processes, or your DataFax administrator, will terminate your **DFsetup** without warning due to some critical situation on your computer network. In either of these situations **DFsetup** will attempt to preserve the current state of your work before exiting.

In the event of an unexpected failure, you will see a warning in your DataFax errorlog similar to the following:

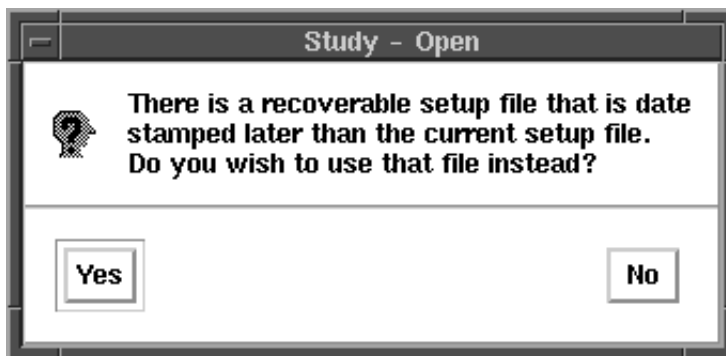
Figure 2.53. Warning message displayed in the event of an unexpected system failure.

```
> DFsetup: an unexpected termination signal has been received.
DFsetup: an attempt has been made to preserve your work...
DFsetup: Study file saved in /clients/s248/lib/DFsetup.preserve.
```

If you have not yet made any changes to your setup, then the sentence 'An attempt has been made to preserve your work' will be replaced by 'Attempting to exit normally....'. In this case, you should simply restart **DFsetup** from your study toolbox.

In the case where you had made changes and **DFsetup** attempted to preserve those changes, you should also restart **DFsetup** from your study toolbox. **DFsetup** will note the existence of a setup preserve file that is more recent than your last saved setup file, and ask you if you wish to load it:

Figure 2.54. A message indicating that a setup preserve file exists.



Click Yes to recover your work as it was when **DFsetup** last exited; otherwise click No to return to the last saved version of the setup file. If you click Yes, it is advised that you immediately review all pages in the setup to confirm that you want to use this version and then select Save from the File menu to write the preserved setup to the setup save file.

2.15.16. Printing a Study Schema

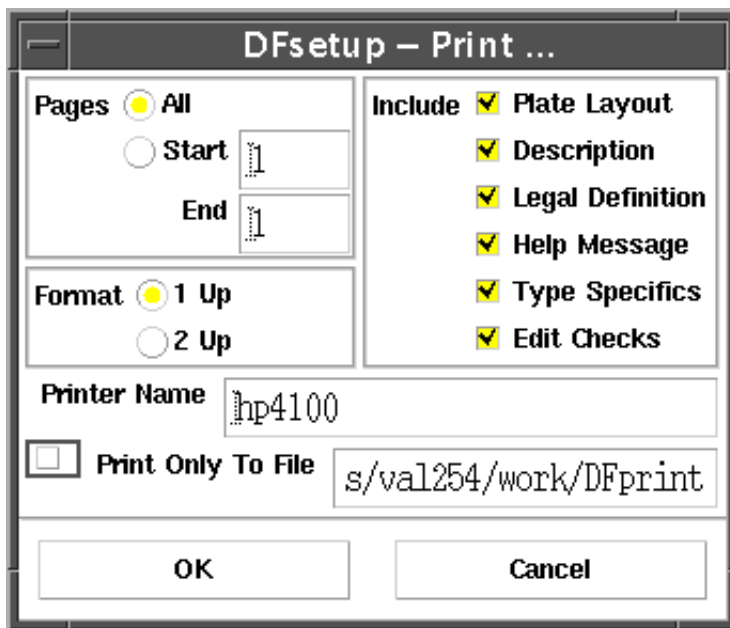
Once plates have been set up and variables defined, you may want to print out a detailed description of the study schema. Printing a study schema can be accomplished in either **DFsetup** using the Print feature in the File menu, or in **DFreports** using the DataFax Generic reports **DF_SSschema** or **DF_SSvars**. For a description of **DF_SSschema**, **DF_SSvars** and their options, see *DataFax Standard Reports Guide, Reference Pages*.

To print the study schema from the Setup Tool:

1. Choose Print from the File menu.

The following Print dialog box will appear.

Figure 2.55. The DFsetup Print dialog box.



2. Set the desired options and provide a printer or file pathname. Select OK to print.
 - **Pages.** This allows you to indicate for which plate or plates you want the schema printed.
 - **Format.** Print format is specified by a toggle button. The 1 Up option will print data in a single column on one side of the page. 2 Up will print data in 2 columns.
 - **Include: Plate Layout.** If Plate Layout is the only option selected, the print-out will display the shape and position of each variable widget on the page. The variable number and Generic name appear within each widget. In addition, a brief description of each variable appears on a subsequent page(s); this includes the variable number, type, unique and generic names, variable style and the value (optional or required).
 - **Include: Description.** Selecting this option will include the variable description defined for each variable on the plates to be printed.
 - **Include: Legal Definition.** If this is selected, the schema print-out will include all legal ranges defined for variables on the plates to be printed.
 - **Include: Help Message.** If selected, the schema print-out will include all defined help messages on the plates to be printed.
 - **Include: Type Specifics.** Selecting this option will include any formatting defined for variables on the selected plates. This includes formatting for any string, numeric and date fields, as well as the coding and labels defined for choice and check fields.
 - **Include: Edit Checks.** If selected, this option will include the name and type (on field enter/exit, on plate enter/exit) of the edit check defined for any variable on the plate(s).
 - **Printer Name.** The name of the printer to be used for printing.
 - **Print only to file.** If you want to save the study schema to a file, select the check box to the left of Print only to file. You must then specify a pathname and filename to which the schema should be printed. By default, DataFax inserts the pathname of the study work directory.

2.15.17. Testing Study Setup

When you exit from **DFsetup** it will automatically create study definition files for any additions or modifications which you have made; including the file map which defines all of the plates known for this setup, the template used by the ICR software to read the data from each CRF plate, and the study database schema which includes all variable definitions for all CRF plates. Once this is done, DataFax will be ready to receive and process CRF faxes for the study, and staff can begin to use all of the DataFax tools to review

them, add quality control notes, etc.

Before you actually start the study it is a good idea to fax in some sample CRFs to test your DataFax setup. This can be done casually or planned and executed rigorously to meet your site's validation requirements. The DataFax Study Setup Worksheets provide checklists for testing your study setup.

Some of the tests that you should consider performing include the following:

1. **Barcodes.** Fax in one copy of each plate and verify that barcodes are positioned and being read correctly.
2. **ICR.** Fax in several examples of each CRF plate and use **DFvalidate** to look for data fields which were consistently read incorrectly across all samples of each plate. Use **DFsetup** to check the positioning and definition of these variables. Make sure that the variable widget is registered on the data boxes, and check the variable type and legal range test.
3. **Skip Patterns.** If you have set skip patterns make sure that they operate as expected and do not interfere with any edit checks.
4. **Legal Value Tests.** Purposely enter illegal values in **DFvalidate** data entry window to make sure that the tests perform as expected.
5. **Quality Control Reports.** Before testing QC reports, you should run the DataFax **DF_ICvisitmap** and **DF_ICschema** reports in the Reports Tool. **DF_ICvisitmap** checks the study visit map for errors, while **DF_ICschema** checks the study schema for errors and inconsistencies. If these reports run without problems, then add some QC notes to the sample CRF plates and validate the test records to at least validation level 1. Then perform the steps required to create QC reports and examine the results to make sure that the reports are created as you expected.

You should check QC reports for the following:

- Were patients included in the correct QC report or assigned to the error monitor?
If not, consult centers database and examine your centers database.
- Were overdue visits and missing plates correctly detected?
If not, consult *DataFax Study Planning Guide, Patient Visit Scheduling* and examine your study visit map.
- If you have specified a study page map did it produce the desired labels?
If not, consult page map and examine your study page map.
- Are you satisfied with the description of data fields on which QC notes were added?
If not, consider modifying the variable Description for those data fields.

2.15.18. DataFax Study Setup Limits

The following crib sheet is a concise and complete listing of DataFax database limits and formats used in study setup. This is to be used in conjunction with the information previously discussed in this chapter as well as with the DataFax Study Setup Worksheets.

Table 2.1. DataFax study setup limits

Description	Limit	Comments
DataFax Study Number	1-255	The suggested range for study numbers is 1-249 as study numbers of 250-255 are reserved for DataFax test and validation studies (e.g. ATK = 254)
Plate Number	1-501	Plates 501 is reserved by DataFax for Quality Control Reports and can not be re-defined at the user level.
Visit/Sequence Number (barcoded)	0-511	
Visit/Sequence Number (first data field)	0-65535	Any data field representing the visit/sequence number, must be defined in the database as field #6.
Center Number	0-2146	This limit applies to the center number only. A patient/subject ID # is concatenated to the center # to obtain the Patient/Subject ID.
Patient/Subject ID Number	0-2147483647	Patient/subject ID numbers are composed of center # + ID #. This limit applies to the concatenated value of the two. The Patient/Subject ID number must always be defined as field #7.
Validation Level (user)	1-7	A user can not be assigned a validation level of 0.
Plate Label (defined using Plate-Edit)	0-100 chars	
Page Map Label (patient status list)	0-18 chars	This limit is achieved by enabling 40 character variable descriptions in the Variables Preferences in DFsetup . Otherwise, the 0-6 character limit is the default.
Page Map Label (refax/Q&A list)	0-32 chars	This limit is achieved by enabling 40 character variable descriptions in the Variables Preferences in DFsetup . Otherwise, the 0-17 character limit is the default.
Visit Label (defined in Visit Map)	1-40 chars	
Variable Style Name	1-15 chars	
Variable Name (unique)	1-80 chars	Unique variable names are required by DataFax
Variable Name (generic)	0-80	Generic names are not required by DataFax
Variable Description	1-40 chars	This limit is achieved by enabling 40 character variable descriptions in the Variables Preferences in DFsetup . Otherwise, the 25 character limit is the default.
Legal Range Definition	0-200 chars	

Description	Limit	Comments
String Field Length (store value)	Limited by overall record length of 4096	
Numeric Field Value	-2147483647 to 2147483647	This limit applies to numeric DataFax field types only.
Check Field	0-65535 (for each code)	A code must be specified for each of “check on” and “check off”.
Choice Field	0-65535 (for each code)	A code must be specified for each choice within a choice field. There must be a minimum of 3 choice codes in order to define the field as a choice field.
Visual Analog Scale (VAS) Minimum and Maximum Endpoint Values	-32768 to 32767	
Visual Analog Scale (VAS) Precision Value	0 to 65535	Precision can not be a negative value.
Date Field Length	6-64 chars	The minimum value of 6 does not include delimiters (e.g. yymmdd).
DataFax Default Date Format	YY/MM/DD	This format applies only when no other format is specified
DataFax Default Implicit Date Range	1920-2019	This range applies if an implicit range is not specified in the Setup Tool
DataFax Date Rounding Default	never	Unless otherwise specified in the Setup Tool, no partial date rounding is performed.
Number of Variables Defined per Plate	1-999	
Maximum Length of a Data Record	4096 chars	For pre-3.4 releases, maximum length of a data record = 2048 chars
Primary Fax Number (defined in centers database)	100 chars	This limit includes all punctuation symbols, and applies to both fax numbers and email addresses.

[3] A default length of 80 characters exists for the Plate Enter/Exit and Field Enter/Exit text fields. If multiple edit check names are required, it may be necessary to increase the default length by changing the application defaults settings of `*.procPreValue.maxLength`, `*.procPostValue.maxLength`, `*.procBeginValue.maxLength`, and `*.procEndValue.maxLength` (see *DataFax User Guide, Customizing appearance and behavior*).

[4] If **iDataFax** is not being used in a study, it is not necessary to publish the edit checks.

Chapter 3. Configuration Tool

3.1. Introduction

DFconfig is a user tool for creating and updating the configuration information that customizes a study setup. Specifically, **DFconfig** can be used to administer the:

- centers database
- visit map
- conditional maps which include termination, cycle, visit, and plate
- page map
- lookup table and missing value maps
- QC and CRF sort map
- QC cover, message, and title text

3.2. Starting DFconfig

The configuration tool can be started in one of two ways:

1. By clicking the **DFconfig** icon in the study toolbox.
2. By choosing the Configuration ... menu item of the Study menu in **DFsetup**.

3.2.1. DFconfig Permissions

Permissions must be granted by the DataFax administrator for any user to access **DFconfig**. The required user permissions are:

- If a user has study permissions for the setup tool (**DFsetup**), then they have update permissions for **DFconfig**. They are permitted to view and modify all parts of the configuration.
- If a user has study permissions for the reports tool (**DFreports**), then they have view-only permissions for **DFconfig**. They are permitted to view all parts of the configuration but they are not permitted to modify anything. In this case, a warning dialog appears when **DFconfig** starts

Figure 3.1. Permissions warning when starting DFconfig



and the tool title is modified to remind the user that they have view only permissions.

Figure 3.2. DFconfig tool title with view only permissions



- A user without the proper permissions receives an error dialog and the tool exits.

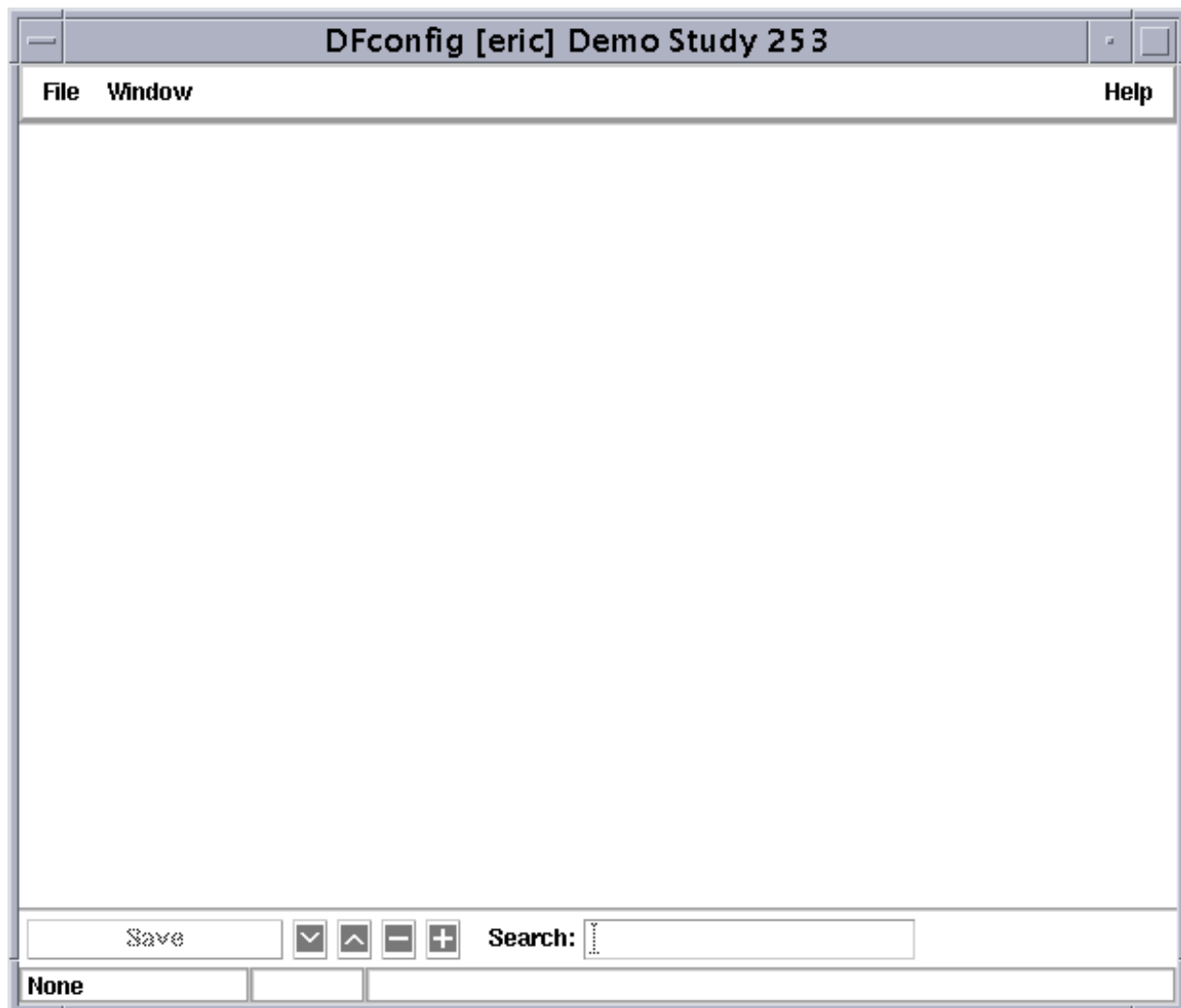
Figure 3.3. Permissions error when starting DFconfig



3.3. DFconfig Window

After initial startup, the main window appears as illustrated in Figure 3.4.

Figure 3.4. The main window of DFconfig



The main window has the following features:

- The top of the main window of **DFconfig** has a title bar that includes the user name and the study name. It may also include the phrase [View only] if the user has view only permissions.
- A menu toolbar is present below the title and above the main area of the interface.
- The main area displays the editor for the chosen configuration. It is blank when the tool starts and before the user chooses a configuration to edit.
- Action buttons reside below the main area.
- At the bottom of the main window is a three panel message line. From left to right, the status message line identifies the type of configuration currently being edited, status flags indicating whether the configuration is already in use and/or whether the configuration is read-only, and messages for any errors encountered.

3.3.1. Choosing a Configuration Window

There are 12 different configurations that can be edited, each has an editor window and unique appearance. At most one editor window can be open at a time. When **DFconfig** starts, no editor is open and the main window is blank.

To choose an editor window, select one from the Window menu. The choices are:

- Centers
- Visit map
- Conditional Termination
- Conditional Cycle
- Conditional Visit
- Conditional Plate
- Page map
- Missing map
- Sort map
- Lookup Tables
- QC Titles
- QC Messages
- QC Covers

Before opening the chosen editor window, the existing editor window, if any, must be closed. If there are unsaved changes, the changes must first be saved or discarded. Once the changes are saved, or if there are no changes to save, the chosen editor window appears in the same main window area replacing the previous window. In the Window menu, the chosen editor is identified with a toggle button. The name of the chosen editor appears in the left-most status line.

To keep **DFconfig** open but close the editor window, choose Window None.

3.3.2. Configuration Window Editors

There are two styles of configuration editor:

1. Spreadsheet editor
2. Text editor.

DFconfig will choose the style automatically based upon the configuration that is being edited.

3.3.2.1. Configuration Editor Common Features

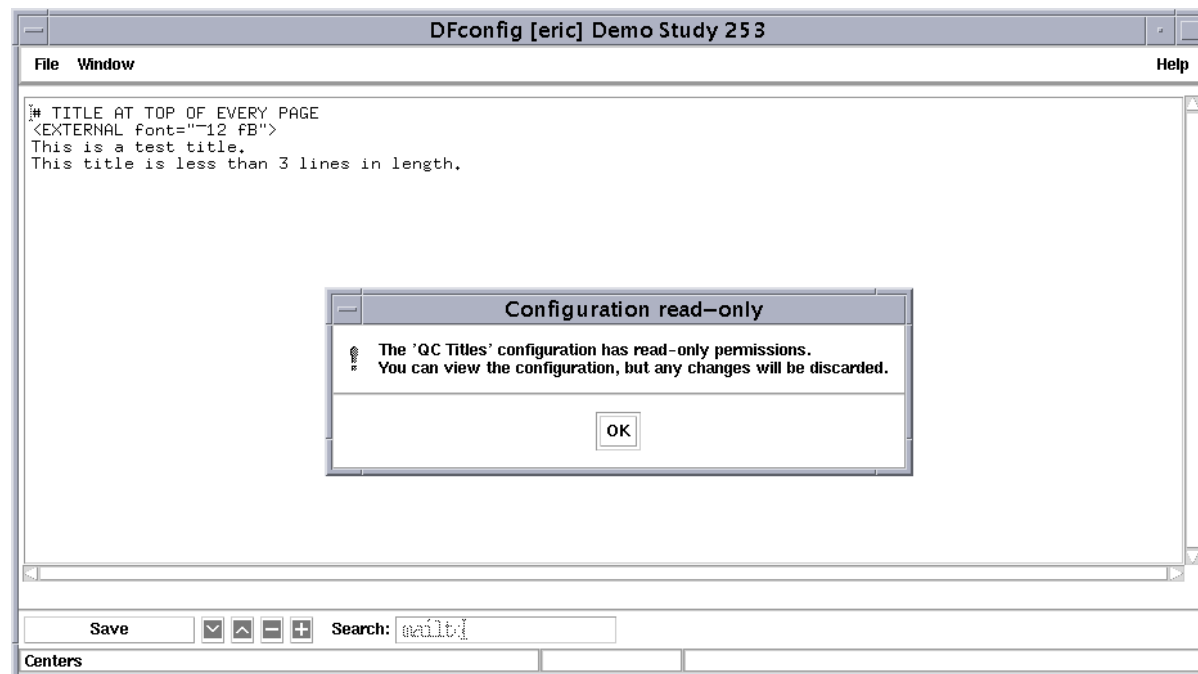
All configuration editors share the following common features:

- **Save (needed).** The left-most button in the action area is labeled Save and is used to save any changes in the current configuration editor. The label is automatically updated to read Save (needed) after any user change, to remind the user that there are unsaved changes. ^[5]
- **View only access.** User permissions on **DFconfig** may have already been set to allow the user view only (read only) access. In this case, the tool title includes [View only] and the Save is disabled for each configuration window. See Figure 3.2 for an example.

It is also possible that only certain configuration windows are marked as view only. ^[6] The creator of these configurations will have set the original permissions to allow or disallow editing.

If the current configuration is view only, a warning dialog, such as that in Figure 3.5 is displayed, and then the acronym +RO appears in the middle pane of the status line, and the Save button is disabled.

Figure 3.5. Read-only access to QC titles



- **File locking.** Each configuration can be edited by at most one person at a time. If the current configuration is being edited by another person, the contents can be viewed but not edited. ^[6] The acronym +LCK appears in the middle pane of the status line and the Save button is disabled. Conversely, while the current window has the configuration open, other users that attempt to edit the same configuration will see the same +LCK acronym and be prevented from editing. For this reason, the current configuration window should be left open only as long as needed.
- **Discarding changes.** There is no action button to discard any unsaved changes. Instead, exiting the current configuration window (either by switching to another window, or closing **DFconfig**) will prompt to save or discard unsaved changes.

3.3.2.2. Spreadsheet Editor

The spreadsheet editor is used when editing:

- Centers
- Visit map
- Conditional Termination
- Conditional Cycle
- Conditional Visit
- Conditional Plate
- Page map
- Missing map
- Sort map
- Lookup Tables

3.3.2.3. Text Editor

The text editor is used when editing:

- QC Titles
- QC Messages
- QC Covers

3.4. Spreadsheet Editor

The spreadsheet editor displays configuration information as rows (displayed horizontally) and columns (displayed vertically), each column holding a category of information (e.g., fax number), and all the columns in one row relating to the same unit of information (e.g., a clinical site). The intersection of a row and a column is a single cell, and a cell holds a single data item.

3.4.1. Common Features

The spreadsheet editor is used to edit varied configuration information. In each application however there are several features of the editor that appear and behave consistently. These features are:

- **focus.** Exactly one cell has the focus. The cell that has the focus receives all keyboard input. Any cell can be given the focus by clicking in it with the mouse.
- **alternating rows/blocks.** Shading is used to offset each row from its neighboring rows. In most configurations a single row is needed to present one unit of information. That is, there is a one-to-one mapping of physical, spreadsheet row to virtual, data row. However in some configurations, notably all of the editors for conditional configurations (termination, cycle, visit, and plate), more than one spreadsheet row is needed to report all of the information for a data row. In this case, the physical rows are grouped together, and shaded, as a block.
- **search.** The contents of the current spreadsheet can be searched for cells that match a search string.

To search for a string:

1. Select the starting point for the search

Searching starts at the focus cell. To search the entire spreadsheet, ensure that the cell in the first column of the first row has the focus.

2. Enter the search string into the Search: field and press Enter

Example 3.1. Searching for the string "mailto"

Search: 1 matches.

Searching starts at the focus cell and moves right across columns in the same row, and then continues by examining each column in the following rows, starting at the first column in each row.

Each cell is examined to see if it contains the search string. The cell value matches the search string if the search string is fully contained in the cell value, using a case sensitive comparison. The cell value and the search string do not need to match identically - the cell value must only contain the search string.

Example 3.2. Search for "Not"

	Code	Label
1	.U	Unknown
2	.D	Not Done
3	.A	Not Available
4	.R	Not Relevant
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		

Search:
3 matches.

- sort.** The information in some columns of some configurations is amenable to sorting. For example, center information can be sorted by ascending center number. Sortable columns are identified with a column label that has a raised, button-like appearance, as shown in Figure 3.6. Clicking on the column label once sorts the rows in ascending order based upon the current cell values in the column. Clicking on the column label again re-sorts in descending order.

Figure 3.6. Column headers that identify sorting

#	Contact Person	Affiliation	Address
---	----------------	-------------	---------

Sorting uses numeric collating sequence for those columns that contain numeric data only (such as center number); otherwise, ASCII collating sequence is used.

**Undo sort action**

There is no undo action and so the only way to undo an unexpected sort result/action is to cancel and discard the changes in the current configuration editor.

- cell validation.** The data in all columns cannot contain the | character or any non-printable character. Each cell prevents the entry of such characters.

The data in some columns is subject to additional syntax and legal limit checks. This validation occurs when the focus leaves the current cell. If the cell is subject to validation, and the value fails validation, the focus will remain on the cell, and a message indicating the validation error appears in the right-most panel of the message line. It is not possible to leave the cell until the validation error is corrected.

3.4.2. Navigation

As previously indicated, exactly one cell has the focus. Any cell can be given the focus (navigated to) by clicking in it with the mouse. It is also possible to navigate to another cell using the keyboard in the following ways:

- **next cell.** Pressing Tab moves the focus to the next cell.
- **previous cell.** Pressing Shift-Tab moves the focus to the previous cell.
- **next row.** Pressing Arrow Down moves the focus to the cell in the same column in the row below the current row.
- **previous row.** Pressing Arrow Up moves the focus to the cell in the same column in the row above the current row.





3.4.3. Row Actions

The spreadsheet editor includes a grouping of action buttons located below the main window area, as shown in Figure 3.7.

Figure 3.7. Action buttons



These buttons are used to perform actions on rows as follows:

-  **move current row down.** To move the contents of the cells in the current row down one row, effectively switching the cells' contents with the row directly below, select the move down action button.
-  **move current row up.** To move the contents of the cells in the current row up one row, effectively switching the cells' contents with the row directly above, select the move up action button.
-  **delete current row.** To delete a row, move the focus to the desired row and select the delete action button. If one or more of the cells is non-blank, a deletion confirmation dialog is presented. If confirmed, the current row is deleted. If the row contains only blank cells, there is no confirmation dialog - the current row is immediately deleted. After deletion, each of the following rows is moved up one to fill the void left by the deleted row.
-  **insert new row.** To insert a new row, move the focus to the row that follows the desired location of the insertion and select the insert action button. A new row is created above the focus row and the focus moves to the first (leftmost) cell of the new row.
- **add new row.** There is no action button to add a new row. To add a row between two rows or before the first row, move the focus to the following row and insert a new row. To add a new row after the last row, move the focus to the last (rightmost) cell of the last defined row and then press Enter. A new row is created and the focus moves to the first (leftmost) cell of the new row.

3.5. Centers Database

The centers database records important information for each investigative center that is participating in the study. It's primary functions are:

- to connect each patient identifier with the center that is responsible for the patient
- to identify the contact fax number(s) and/or email addresses for delivery of QC reports to the center

Additional information may also be recorded but that information is not essential for **DataFax** operation.

Example 3.3. Centers database

[illegible]

The following information is available for each center. The information that is required is identified as such. The DataFax report **DF_ICcenters** should be executed to verify the consistency of the required information. This report can be executed using the Check Consistency option in the Window menu or from the DataFax **DFreports** tool. Further details can be found in *DataFax Programmer Guide, DFcenters - centers database*.

#. The center number is required. Each value must be numeric, unique for the study, and in the range 0 to 2146, inclusive.

Contact Person. The contact person is required. It typically contains the full name of the primary contact person at the center, but it may contain any text.

Affiliation. The affiliation is required. It is typically the name of the center itself or the institution that the center is affiliated with.

Address. The address information is optional. If specified, it should include the full postal address of the center.

Primary Fax. The primary fax is required. Despite its historical name, this field may contain one or more fax numbers and/or email addresses, each value separated from the others by a space or a comma. When a QC report is sent to the center via the **DF_QCfax** report, this information is used to deliver the report. Each fax number must contain all of the digits required to dial the destination fax machine, including long distance code, country code, and area code, if any. Spaces within the number are not permitted as they identify a break between multiple, adjacent numbers, however other punctuation characters such as (,), and – are allowed and may aid readability. Each email address must be the complete email address needed to reach the destination mailbox, plus the required `mailto:` prefix. This prefix is required for each email address.

Secondary Fax. The secondary fax information is optional. It should include any other fax numbers that the center may have. **DataFax** will *not* use this number to deliver information to the center if the primary number(s) fail.

Phone. The phone information is optional. If specified, it should include the complete number needed to reach the center by telephone.

Investigator. The investigator is optional. It is the name of the investigator at the center. Note that QC reports sent to the center are addressed to the contact person and not the investigator.

Inv. Phone. The investigator phone is optional. If specified, it should include the complete number needed to reach the investigator by telephone.

Reply To. The reply to value is optional, but is highly recommended for centers that receive their QC reports via email. The value is a complete email address of a member of the central data management staff (unlike the Primary Fax field, the `mailto:` is not expected). It appears as the `From:` address whenever the QC report is successfully delivered to an email address specified in the Primary Fax field. This provides a real email address that the center can use to direct any questions to. If the QC report is sent via email and this field is blank, any reply emails from the center will be delivered to the DataFax problem mail recipient defined by the **DataFax** administrator. If a QC report transmission fails, the report will also be re-directed to the DataFax problem mail recipient.

Patients. The patient identifier(s) value is required. It identifies the values and/or ranges of patient identifiers that the center is responsible for. The syntax is the standard **DataFax** list syntax:

- a comma is used to separate one list from another
- each item may be a single numeric patient identifier or a range of patient identifiers expressed as the minimum value of the range, the range delimiter (–), and the maximum value of the range. Exactly one center must be identified in this column as the error monitor, as described in Section 3.5.1.



Modifying the Centers Database

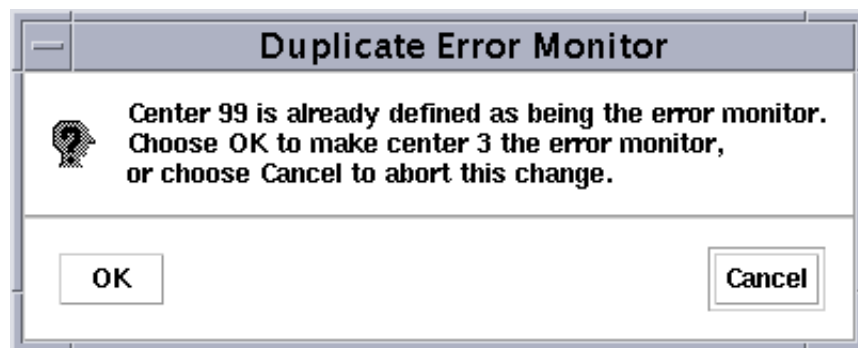
The centers database may be modified at any time for an active study. If quality control reports are being created for the study, the DataFax report **DF_QCupdate** must always be run after **DFcenters** modification and prior to QC report creation with **DF_QCreports** to ensure that the most up-to-date **DFcenters** file will be used when generating patient status and scheduling information for the QC reports. Failure to run **DF_QCupdate** may result in errors and/or omissions in the QC reports.

3.5.1. Error Monitor

Exactly one of the centers must be designated as the *Error Monitor*. The error monitor receives unresolved, external QC notes for all patients not already attributed to other centers.

The error monitor designation is made in the currently blank cell of the Patients column, simply by entering e or E (lower- or upper-case letter e). The input will be automatically completed to read Error Monitor. If the cell is not blank, the cell must first be blanked.

If another center has already been designated as the error monitor, the following warning will appear before any change is made. The dialog reports the center number that is currently the error monitor and the new center number.



If OK is selected, the value in the Patients cell for the previous center number is blanked and Error Monitor is inserted in the current cell. Otherwise, if Cancel is selected, no change is made.

3.6. Visit Map

The visit map contains the key information needed for patient scheduling. This, along with the various conditional maps, are used to schedule and manage all patient assessments, and ensure that the required data is received in a timely fashion. The visit map is comprised of 2 levels of organization,

1. **Cycles.** Cycles act as a container for one or more patient visits. There are 3 types: screening, in-study, and end cycles.
2. **Visits.** Visits represent the actual in-clinic visits or assessments that each patient undergoes.

A detailed description of cycles, visits, and the rules governing their definition, can be found in *DataFax Study Planning Guide, Patient Visit Scheduling*.

3.6.1. Visit Map Editing

The study visit map is presented in a combination view using the spreadsheet editor and also a detail pane. The spreadsheet allows entry and modification of the visit or cycle number as well as the visit or cycle label. The spreadsheet also displays the current settings for the visit/cycle type, the due day and the number of days overdue allowance. These values can be entered and modified in the detail pane. The detail pane is also used for editing all of the other information related to a visit or cycle.

3.6.2. Cycle Definition

A visit map consists of one or more cycle records, each followed by a series of visits scheduled to occur within the cycle. In the visit map editor, each cycle record is denoted by C: *cycle type* and unique coloring of the cycle row in the spreadsheet.

To define a new cycle:

1. Add/insert a new row in the spreadsheet

The position of the cycle record is important to its meaning. All following visit records, up to the next cycle record, are deemed to belong to this cycle.

2. Assign a unique cycle number

Enter a unique number for the cycle in the column V# or C# of the spreadsheet view. The screening cycle must always be assigned a cycle number of 0, while all in-study cycles must be numbered consecutively within the visit map, beginning at 1. The end cycle must be assigned the next consecutive number following the last in-study cycle.

3. Specify a cycle label

A cycle label is entered in the Label column of the spreadsheet view and consists of descriptive text, no more than 32 characters in length.

4. Set cycle and type

In the detail pane, select the Cycle toggle button. From the pull-down list located below the button, select the desired cycle type from

- C: Screening Cycle
- C: In-Study, Conditional Cycle
- C: In-Study, Required Cycle
- C: In-Study, Optional Cycle
- C: End Cycle

5. Define cycle scheduling

Specify when the first visit in the cycle is due (in number of days), and indicate from which of the following the cycle should be scheduled

- Visit Number
- Baseline, First Cycle
- Baseline, Previous Cycle
- Term, Previous Cycle
- Defined Condition
- Not Scheduled

If the cycle is to be scheduled from a defined condition, the scheduling condition must be entered using the conditional cycle editor. The cycle due day is only relevant for in-study cycles.

6. Define when the cycle is overdue

Indicate the number of days at which the cycle should be considered past due if it has not already taken place. An overdue allowance is only relevant for in-study cycles.

The following example shows a visit map consisting of 5 cycles - 1 screening cycle, 3 in-study cycles, and 1 end cycle. The numbering assigned to the cycles is consecutive within the visit map, beginning at 0 for screening and ending with 4 for the end cycle. Within each cycle, one or more visits are defined with their own unique visit number and are scheduled to occur at different points in time.

Example 3.4. Cycle definition

DFconfig [eric] Demo Study 253

File Window Help

Type	Due Day	+Allow	V# or C#	Label
C: Screening Cycle			0	SCREENING
X: Screening	0	0	91	Screening Visit 1
X: Screening	-4	2	92	Screening Visit 2
C: In-Study, Required Cycle	3	5	1	TREATMENT
B: Baseline			0	Rx Step 1
S: Scheduled	30	3	1	Rx Step 2
T: Termination	60	5	2	Rx Evaluation
C: In-Study, Conditional Cycle	7	7	2	RE-TREATMENT
B: Baseline			3	Repeat Rx Step 2
T: Termination	30	3	4	Re-Rx Evaluation
C: In-Study, Required Cycle	30	7	3	FOLLOW-UP
B: Baseline			10	1 Month Post-Rx
O: Optional			11	6 Months Post-Rx
T: Termination	335	7	12	1 Year Post-Rx
C: End Cycle			4	REPORTS
O: Optional			101-199	AE Report #%(S.2.2)

☐ Visit ☒ Cycle
 C: In-Study, Required Cycle ☐
 Scheduling
 Cycle is due: days from visit
 or from ☐
 The cycle is overdue when days late

Save ☒ ☐ ☐ ☐ Search:

Visit map

3.6.3. Visit Definition

The study assessments that comprise each cycle are entered in the visit map editor following their respective cycle record. Each cycle consists of one or more visit records which is denoted by the entry *visit type: visit label*.

To add a visit to the visit map:

1. Set visit and type

Add a new row to the visit map editor. With the new row as the current entry, select the Visit button. From the pull-down list located below the button, select the desired visit type from

- S: Scheduled
- T: Termination
- R: Required by Study End
- O: Optional
- E: Early Termination
- B: Baseline
- P: Pre-Baseline
- r: Required by Next Visit
- X: Screening
- A: Abort All Follow-up
- W: Time Window Termination
- F: Final

2. Specify a visit date field

Visit dates represent the date on which the assessment occurred. Select a visit date plate and field for the current visit from the pull-down menu. The list of visit dates presented to the user represent those date fields that have been defined with the VisitDate style in **DFsetup**. The visit date selected, must be defined on a required plate for the current visit.

3. Define visit scheduling

Visits are scheduled relative to the baseline visit of the current cycle. The relative schedule is expressed as a number of days before or after the baseline visit date.

Alternatively, visits can be scheduled in a termination window. The termination window applies if the visit type is W: Time Window Termination. In such studies, a definition of how the final visit will be scheduled is needed. DataFax includes the following methods:

- **on a specific date.** For example, on 1999/12/01. This schedules the final followup visit on Dec 1, 1999 for all patients.
- **before a specific date.** For example, before 1999/12/01. The last scheduled follow-up visit for each patient before Dec 1, 1999 becomes the termination visit.
- **after a specific date.** For example, after 1999/12/01. The first scheduled followup after Dec 1, 1999 becomes the termination visit.
- **between two specific dates.** For example, between 1999/11/01~1999/11/30 .25. All final follow-ups are to be scheduled in November 1999. The scaling factor (.25 in the above example) spreads the visits out over the termination interval. It should be set to the termination window width (days) divided by the usual inter-visit interval (days). In the above example a scaling factor of .25 would be appropriate for the 1 month termination window if the usual gap between follow-up visits is 4 months.

4. Define when the visit is overdue

The visit is overdue when a specified number of days has passed after the due date.

5. Specify required, optional and missed visit plates

For each visit, one or more plates are required. Enter these plates as a single value, list of values, or range of values. The plate which includes the visit date field must always be included.

Within a visit, other plates may be optional. Enter these plates as a single value, list of values, or range of values. It is also possible to specify that some plates are conditionally required or conditionally excluded (not expected). This is specified in conditional maps which include termination, cycle, visit, and plate.

If the visit includes a plate that is to be submitted when the visit does not take place, specify the missed visit plate number.

A completed visit definition looks like Example 3.5.

Example 3.5. Visit definition

DFconfig [eric] Demo Study 253				
File Window				Help
Type	Due Day	+Allow	V# or C#	Label
C: Screening Cycle			0	SCREENING
X: Screening	0	0	91	Screening Visit 1
X: Screening	-4	2	92	Screening Visit 2
C: In-Study, Required Cycle	3	5	1	TREATMENT
B: Baseline			0	Rx Step 1
S: Scheduled	30	3	1	Rx Step 2
T: Termination	60	5	2	Rx Evaluation
C: In-Study, Conditional Cycle	7	7	2	RE-TREATMENT
B: Baseline			3	Repeat Rx Step 2
T: Termination	30	3	4	Re-Rx Evaluation
C: In-Study, Required Cycle	30	7	3	FOLLOW-UP
B: Baseline			10	1 Month Post-Rx
O: Optional			11	6 Months Post-Rx
T: Termination	335	7	12	1 Year Post-Rx
C: End Cycle			4	REPORTS
O: Optional			101-199	AE Report #%(S.2.2)

Save

☒ ☐ ☐ ☐

Search:

☒ Visit
☐ Cycle

T: Termination

Visit Date Field

Plate 5, field 9=Visit Date

Scheduling

Visit is due: days after baseline
 or in the window
 and is overdue when days late

Plates

Required
 Optional
 Missed Visit

Visit map

3.6.4. Check Visit Map Consistency

It is always recommended to check the consistency of a visit map definition when it, or any conditional map definition, has been created or changed. The consistency of information in **DFcenters** should be checked simultaneously with the visit map definitions as problems with **DFcenters** entries may have an impact on communication with the sites via DataFax. To encourage frequent consistency checking, it has been added to the **DFconfig** tool. This is equivalent to execution of the **DF_ICvisitmap**, **DF_ICschema**, and **DF_ICcenters** reports in the **DFreports** tool.



Important

It is extremely important to ensure that the visit map definition has no consistency check errors. Otherwise, the validity of the study QC report is not assured.

To check the consistency of the visit map definition,

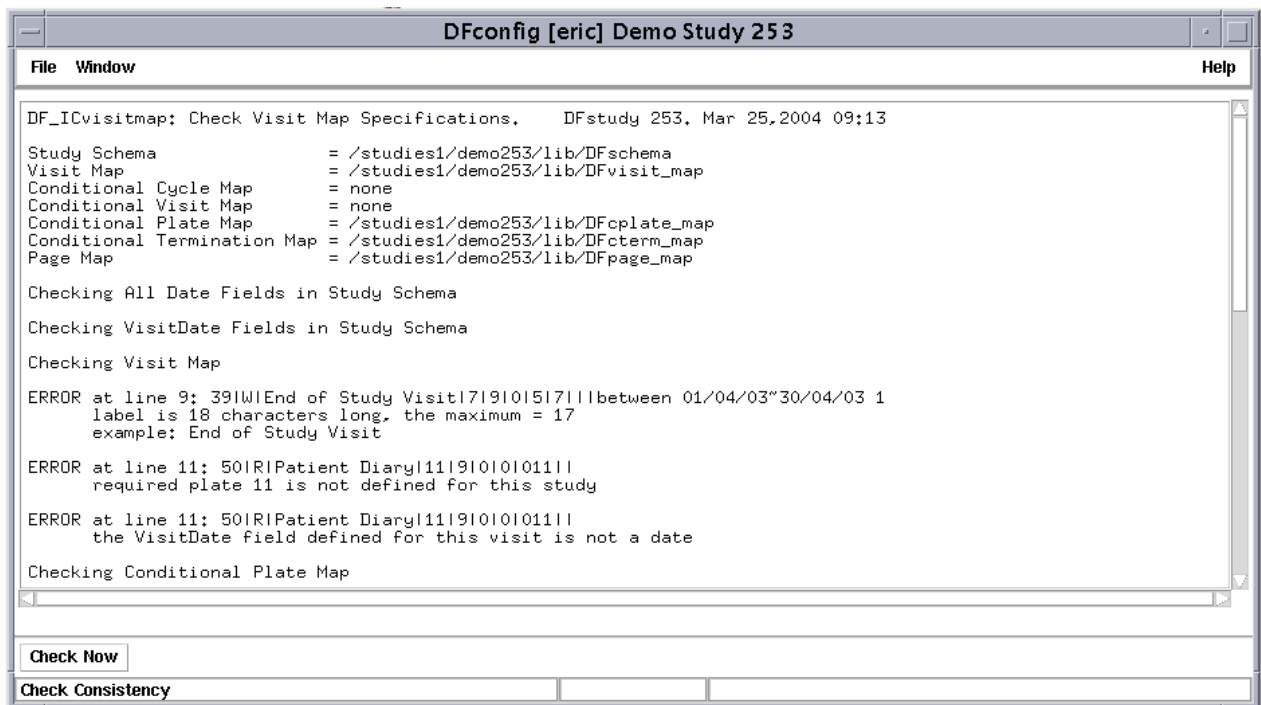
1. Switch to the check consistency view

From the Window menu, choose > Check Consistency. If there are any unsaved changes in the current editor, a confirmation dialog to save will appear.

- ## 2. Select Check Now

Consistency checking starts. Any errors and/or warnings are reported in the text output window.

Example 3.6. Check consistency



3.7. Conditional Maps: Termination, Cycle, Visit, and Plate

The conditional maps extend the functionality of the visit map by allowing items to be conditionally required or conditionally excluded. This can occur at the cycle level (conditional cycle map), the visit level (conditional visit map), and/or the plate level (conditional plate map). In addition, patient schedule termination can be conditional and specified using the conditional termination map.

Details regarding the purpose of conditional maps can be found in *DataFax Study Planning Guide, Conditional Cycle Map*, *DataFax Study Planning Guide, Conditional Visit Map*, *DataFax Study Planning Guide, Conditional Plate Map*, and *DataFax Study Planning Guide, Conditional Termination Map*.

3.7.1. Defining a conditional map

All conditional maps are constructed from one or more conditional records. Each conditional record has the following characteristics:

- a condition expression. The condition expression always involves a field on a plate of a visit. The value in the field equals the condition. The condition expression may also be compound, involving two or more expressions.
- one or more actions to execute if the condition is true.

3.7.2. Defining a conditional termination

A study may have 0 or more conditions that lead to early termination. A condition is met when a particular field equals a value. When the condition is met, the termination can specify that the current cycle is terminated or that the remainder of the patient is terminated.

In this example, a conditional termination has already been defined, in row 1, which states if field 9, on plate 1 or visit 0 equals the value 2, then all follow-up is aborted. A new conditional termination is being defined. Start a new row in the spreadsheet. The first step is to enter the condition expression in the first 5 columns. Enter the letter **i** in the first column - the editor will automatically complete this to read **IF**. The condition is true if field 9 on plate 27 for any visit in the range 1000 to 1240, inclusive, has a value greater than 5000.

	Field	on Plate	at Visit(s)	equals	then
1	IF	9	1	0	2
2	IF	9	27	1000-1240	>5000

If the condition is true, then the current cycle is terminated. To enter this in the final column, enter the letter **t** - the editor will automatically complete the cell to read **terminate the current cycle**. Alternatively, to include that all follow-up should be aborted, enter the letter **a** - the editor will automatically complete the cell to read **abort all follow-up**. The definition of the conditional termination is complete.

	Field	on Plate	at Visit(s)	equals	then	
1	IF	9	1	0	2	abort all follow-up
2	IF	9	27	1000-1240	>5000	terminate the current cycle

In another example, the condition is a compound condition. The condition is true if field 9 on plate 27 for any visit in the range 1000 to 1240, inclusive, has a value greater than 5000, and field 10 on plate 27 for any visit in the range 1000 to 1240, inclusive, is equal to 1 or 2.

Start a new row in the spreadsheet. Enter letter **i** in the first column, 9 in the second column, 27 in the third column, 1000-1240 in the fourth column, and >5000 in the fifth column. Start a second new row in the spreadsheet, immediately below the first one. Enter letter **a** in the first column (the editor will automatically complete the cell to read **AND**), 10 in the second column, 27 in the third column, 1000-1240 in the fourth column, and 1, 2 in the fifth column.

	Field	on Plate	at Visit(s)	equals	then
1	IF	9	1	0	2
2	IF	9	27	1000-1240	>5000
	AND	10	27	1000-1240	1,2

To complete the conditional termination, the action must be specified in the sixth column. Move the cell focus to the sixth column of the first row of the condition definition. Enter the letter **t** to indicate that only the current cycle is terminated or the letter **a** to indicate that all of the follow-up is aborted. The definition of the conditional termination is complete.

	Field	on Plate	at Visit(s)	equals	then
1	IF	9	1	0	abort all follow-up
2	IF	9	27	1000-1240	>5000
	AND	10	27	1000-1240	1,2

3.7.3. Defining a conditional cycle

A conditional cycle definition requires a condition (or compound condition) and an action that indicates if cycles are required, excluded, or optional. The condition is defined in the same manner as already shown for conditional terminations. To specify the action, enter a cycle number, list of cycles, or range of cycles in the sixth column and the letter *r* (required), *e* (excluded), or *o* (optional) in the seventh and last column. The editor will automatically complete the value in the seventh field.

In this example, the condition is true if field 9 on plate 42 of visit 2000 equals the value 2. If the condition is true, the actions specify that cycle 2 is required, cycle 0 is excluded (not expected), and cycle 3 is optional.

	Field	on Plate	at Visit(s)	equals	then, Cycle(s)	are
1	IF	9	42	2000	2	required
					0	excluded
					3	optional

3.7.4. Defining a conditional visit

A conditional visit definition is identical to a conditional cycle definition, except that the action(s) specify visits rather than cycles.

Example 3.7. Conditional visit map

DFconfig [eric] Demo Study 253							
File Window							He
	Field	on Plate	at Visit(s)	equals	then, Visit(s)	are	
1	IF	11	5	10	>95	11	required
2	IF	13	5	10	>95	11	required
3	IF	6	9	*	>101	101-value	required
4	IF	28	5	*	>0	101-100+value	required
5	IF	17	1	91	>104	92	optional
						93	excluded
						94-99	required
6	IF	19	1	91	>104	92	optional

3.7.5. Defining a conditional plate

A conditional plate definition is similar to a conditional visit definition, but it requires one additional specification: the plate number(s) for the action(s).

In this example, the conditional plate specification has three conditions. Add three consecutive new rows to the spreadsheet. In the first row, enter the first condition. Enter the letter *i* in the first column, 10 in the second column, 2 in the third column, 0 in the fourth column, and 2 in the fifth column. In the second row, enter the second condition. Enter the letter *a* in the first column, 11 in the second column, 2 in the third

column, 0 in the fourth column, and 1 in the fifth column. In the third row, enter the third condition. Enter the letter a in the first column, 12 in the second column, 2 in the third column, 0 in the fourth column, and 1 in the fifth column.

	Field	on Plate	at Visit(s)	equals	then, Plate(s)	at Visit(s)	are
1	IF	10	2	0	2		
	AND	11	2	0	1		
	AND	12	2	0	1		

If all of the conditions are true, then the actions are executed. The first action specifies that plate 5 at visit 0 becomes required.

	Field	on Plate	at Visit(s)	equals	then, Plate(s)	at Visit(s)	are
1	IF	10	2	0	2	5	0 required
	AND	11	2	0	1		
	AND	12	2	0	1		

In addition, plates 10, 11, and 12 at visit 1 become excluded.

	Field	on Plate	at Visit(s)	equals	then, Plate(s)	at Visit(s)	are
1	IF	10	2	0	2	5	0 required
	AND	11	2	0	1	10-12	1 excluded
	AND	12	2	0	1		

Finally, plates 6 through 10 inclusive at visit 0 are optional, and plates 40 and 42 at visit 2 become excluded. The third action can be entered next to the third row's condition. However, the fourth row requires an additional row in the spreadsheet. Add a new row after the fourth row. The first five columns are left blank. The fourth action can then be entered in columns six, seven, and eight.

	Field	on Plate	at Visit(s)	equals	then, Plate(s)	at Visit(s)	are
1	IF	10	2	0	2	5	0 required
	AND	11	2	0	1	10-12	1 excluded
	AND	12	2	0	1	6-10	0 optional
					40,42	2	excluded

3.8. Page Map

The page map allows custom labels to be defined for combinations of visits and plates. These custom labels override the default labels that DataFax uses to identify CRFs in QC reports and PDF exports. Additional information regarding the purpose of the page map can be found at *DataFax Study Planning Guide, CRF Page Map*.

The page map is maintained with the spreadsheet editor, as shown in Example 3.8. In fact, it is maintained with two spreadsheet editors of similar purpose but for slightly different information. Patient status labels appear in the patient status summary of every QC report. They identify the label to be used in a visit number reference. QA/Refax list labels appear in the body of the QC report. They identify the label to be used to identify each CRF that a QC note is attached to.

Example 3.8. Page map

DFconfig [eric] Demo Study 253

File Window Help

Plate/Visit Title

VISIT & CRF FORM

Patient Status Labels

	Visit(s)	Label
8	10	1 Month
9	11	6 Month
10	12	1 Year
11	40	Death
12	101-199	AE#%{S.2.2}

QA / Refax List Labels

	Visit(s)	Plate(s)	Label
14	10	6	Month 6 Form 6
15	10	6	Year 1 Form 6
16	*	8	Form 8, Page %{S.1}
17	*	9	Form 9, Page %{S.2}
18	40	10	Death Report

Save [checkbox] [up] [down] [plus] Search: [text box]

Page map

The visit number and/or plate number column may contain:

- a single value
- a value range
- a list of values, separated by commas
- the literal * (asterisk), meaning all values

The label column contains a text label. The text label will appear whenever a CRF, with keys matching the values in the plate and visit number columns, is referenced. The text label may contain references to data values as described in *DataFax Study Planning Guide, Using Data Values in Page Map Labels*.

Order of appearance for each row is important in each spreadsheet. If more than one row matches the visit and plate number for a CRF/QC note, the row which appears earlier in the spreadsheet is used.

3.9. Lookup Table and Missing Value Maps

Lookup table and missing value map definitions are both maintained with the spreadsheet editor, and are very similar to each other. The definition of a row in each requires two column values. In the lookup table editor, as shown in Example 3.9, each row requires the symbolic table name and the filename of the physical, underlying file to be tied to the table name. The filename may be an absolute path to a file (better for tables shared among studies) or a relative path, in which case it is relative to the study lib directory (better for tables unique to the study).

Example 3.9. Lookup table map

DFconfig		
File Window		
	Table name	File name
1	æcodes	COSTART.lut
2	drugs	DRUGS.lut
3	drugind	DRUGIND.lut
4	QC	QCquery.lut
5	investigators	ISIG.lut
6	REASON	DFreason.lut
7		
8		
9		
10		
11		

In the missing value map, an example of which is show in Example 3.10, each row requires the missing value code and a meaningful label matching the missing value code. If the study data will eventually be exported to SAS, ensure that the chosen missing value codes are also meaningful and legal in SAS.

Example 3.10. Missing value map

DFcon		
File Window		
	Code	Label
1	.U	Unknown
2	.D	Not Done
3	.A	Not Available
4	.R	Not Relevant
5		
6		
7		

There is no implied sorting of rows in either map. The rows may be re-sorted by either column value by clicking on the corresponding column label, but the sorting is for spreadsheet editor presentation purposes only.

If a missing value map is not defined for a study, the DataFax default missing value code, * (asterisk), will be available and can be assigned to any data field when working in **DFvalidate**. It is possible to override this default either by defining a missing value map with different missing value codes, or by saving a missing value map that has no entries at all. Saving an empty missing value map indicates that no missing values are permitted, and makes it impossible to assign missing value codes in **DFvalidate**. If this is the desired behavior, an empty missing value map should be created at the beginning of the study before any missing value codes have been entered.

Changing the missing value map after some of the missing value codes have already been used will result in those changed or deleted codes no longer being recognized as identifying missing values, and instead they will be treated as normal data values. Further, for some data fields, those data values may now represent invalid or illegal responses. Missing value codes rendered invalid can be identified using **DFcmp-Schema** and can be removed from the database by re-validating the affected data records.

3.10. QC and CRF Sort Map

The order of appearance within patient of retrieved CRFs in **DFviewer** and **DFvalidate** as well as the order of appearance of QC notes in formatted QC reports is defined by an optional QC and CRF sort map, often just called **DFqcsort**.

Sorting is always done by ascending patient id number. Within a patient however, sorting can be controlled with the sort map. Without the sort map, within patient sorting is by ascending visit number, and by ascending plate number within visit number. The sort map can be used to alter this. For additional details, consult *DataFax Standard Reports Guide, Customized QC note sort order*,

3.10.1. Editing the QC and CRF Sort Map

Editing **DFqcsort** is done through a spreadsheet editor, an example of which is shown in Example 3.11. Each row defines the sort priority for a single combination of visit and plate numbers. The visit number and/or plate number column may contain:

- a single value
- the literal * (asterisk), meaning all values

The order column may contain any integer number greater than 0. The lower the value in the order column, the earlier that row sorts in the sort sequence. For example, the order value 12 sorts the visit and plate defined in its row ahead of the order value 20. The sort order for two rows that have the same value in the order column is not defined.

Each column in the editor can be sorted for presentation by clicking on the column label. Note however that the actual sort order used for QCs and CRFs is as specified by the order column, which is independent of the presentation order. It may simplify understanding to keep the presentation order sorted by ascending order value so that the two are in sync.

Example 3.11. QC and CRF sort map

In this example, visit 5, plate 8 is given the highest sort priority and hence always sorts to the beginning of patient output. Visit 6 (all plates) sorts after visits 5, 11-15, 21-25, and 91, but ahead of all other visits.

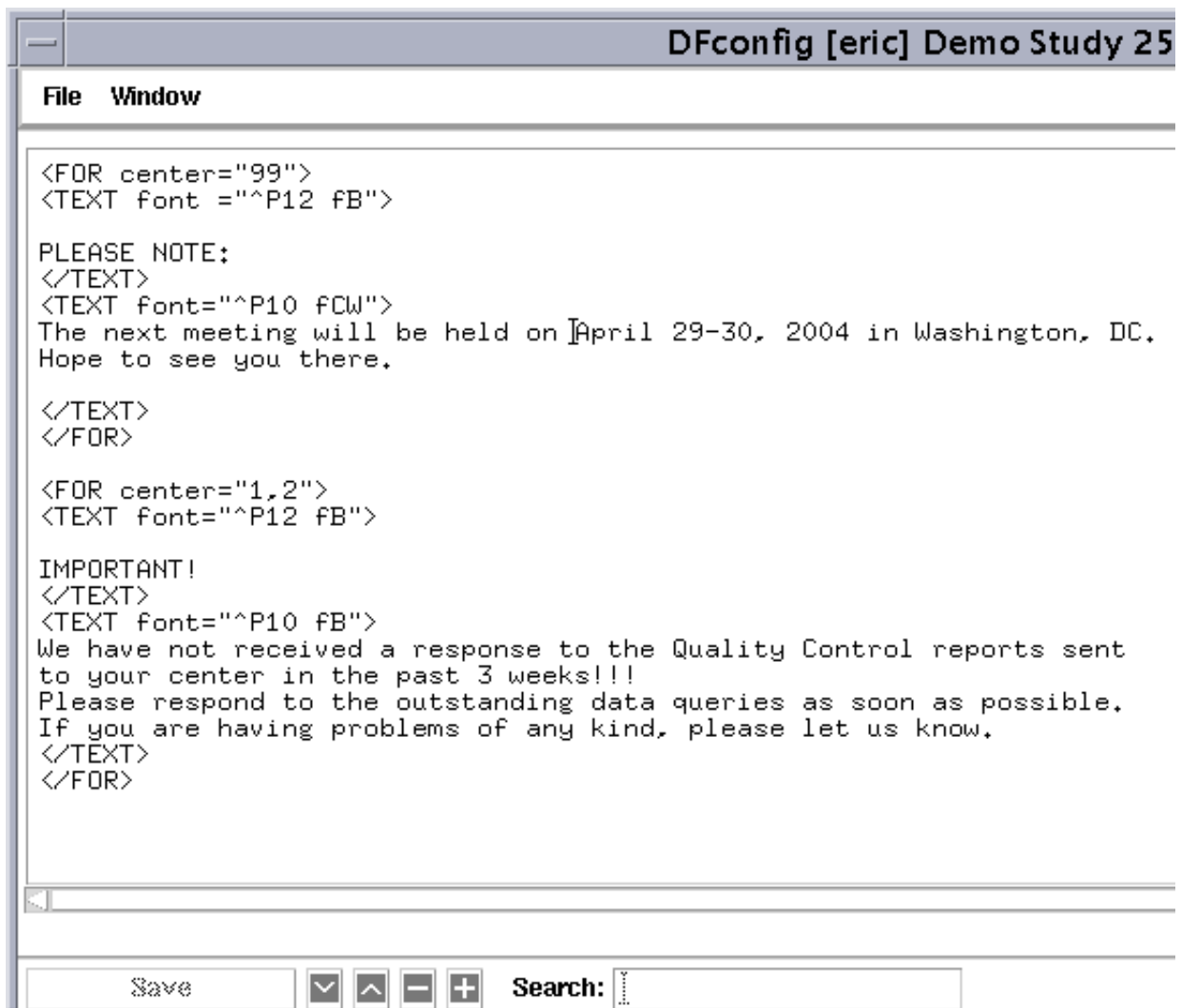
File Window			
	Visit	Plate	Order
11	5	8	10
12	5	14	11
13	5	19	12
14	11	*	21
15	12	*	22
16	13	*	23
17	14	*	24
18	15	*	25
19	21	30	31
20	22	30	32
21	23	30	33
22	24	30	34
23	25	30	35
24	91	40	40
25	6	*	50
26	*	*	99

3.11. QC Cover, Message, and Title Text

It is possible to customize the messages and titles used in a QC report and also to include a customized cover sheet. The purpose of these customizations is described in *DataFax Standard Reports Guide, QC Report Titles*, *DataFax Standard Reports Guide, Cover sheets*, and *DataFax Standard Reports Guide, Messages*.

The editor for these QC report customizations is the text editor. Changes to the text can be typed directly into the window. The arrow keys can be used to move the text cursor as can the mouse. All changes should be saved before switching to another editor window.

Example 3.12. QC Messages



3.12. Exiting DFconfig

To exit **DFconfig**, choose File Exit or File Close. **DFconfig** will close.

If there are unsaved changes in the current editor when exiting, a warning dialog appears.

Figure 3.8. Warning dialog for unsaved changes



Save the changes and exit by choosing Save changes, discard the changes and exit by choosing Discard changes, or cancel the exit and return to the current editor by choosing Cancel.

3.12.1. *Exit status*

If started from a command-line, **DFconfig** will exit with status 0 if the program exited without error. Otherwise, the exit status will be greater than 0.

^[5] Once a change is made and the label is updated to Save (needed), the label will not change back to Save until the changes are saved or discarded. This is true even if the user edits the configuration to undo their unsaved edits.

^[6] The view only and file locking attributes of the configuration editor are evaluated when the editor opens. If these attributes subsequently change in value, the editor will not be aware of the change until it is restarted. This can be achieved by exiting and restarting **DFconfig** completely, or more simply, by switching to another configuration window and then switching back.

Chapter 4. Status Tool

4.1. Introduction

The Status Tool (**DFstatus**) provides a quick check on the current status of a study. It can be used to check the current activity of all users who are working on the study database, determine if there are any new CRF pages awaiting validation, and produce a summary report showing the current validation level and status of all records in the study database. The following example shows a status check on a DataFax study named Test Study, by a user with the login name samer, on Tuesday, July 25, 2006.

Figure 4.1. The DFstatus window.

DFstatus [samer] Test Study																																																																																																			
File Actions									Help																																																																																										
✓ Server Status DataFax Master is running on <i>apache</i> Study Server is running on <i>apache</i> Number of records awaiting validation 0 Number of records being validated 0 Number of records pending iDataFax entry 6																																																																																																			
✓ User Status <table border="1"> <thead> <tr> <th>User</th> <th>Display</th> <th>Tool</th> <th>On Since</th> <th>Locks</th> </tr> </thead> <tbody> <tr> <td>datafax</td> <td>dhcp214.datafax</td> <td>Admin Tool</td> <td>Tue Jul 25 12:38:36 2006</td> <td>no</td> </tr> <tr> <td>samer</td> <td>apache:0.0</td> <td>Status Tool</td> <td>Tue Jul 25 12:32:34 2006</td> <td>no</td> </tr> <tr> <td>david</td> <td>dhcp214.datafax</td> <td>iDataFax</td> <td>Tue Jul 25 12:37:46 2006</td> <td>yes</td> </tr> </tbody> </table>										User	Display	Tool	On Since	Locks	datafax	dhcp214.datafax	Admin Tool	Tue Jul 25 12:38:36 2006	no	samer	apache:0.0	Status Tool	Tue Jul 25 12:32:34 2006	no	david	dhcp214.datafax	iDataFax	Tue Jul 25 12:37:46 2006	yes																																																																						
User	Display	Tool	On Since	Locks																																																																																															
datafax	dhcp214.datafax	Admin Tool	Tue Jul 25 12:38:36 2006	no																																																																																															
samer	apache:0.0	Status Tool	Tue Jul 25 12:32:34 2006	no																																																																																															
david	dhcp214.datafax	iDataFax	Tue Jul 25 12:37:46 2006	yes																																																																																															
✓ Database Status <div> Centers <input type="button" value="..."/> <input type="text"/> Patient Ids <input type="text"/> Visits <input type="button" value="..."/> <input type="text"/> Plates <input type="button" value="..."/> <input type="text"/> </div> <table border="1"> <thead> <tr> <th>Level</th> <th>Clean</th> <th>Dirty</th> <th>Error</th> <th>CLEAN</th> <th>DIRTY</th> <th>ERROR</th> <th>Lost</th> <th>Total</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>12</td> <td>53</td> <td>0</td> <td>343</td> <td>545</td> <td>0</td> <td>1</td> <td>954</td> </tr> <tr> <td>2</td> <td>17</td> <td>0</td> <td>0</td> <td>108</td> <td>36</td> <td>0</td> <td>0</td> <td>161</td> </tr> <tr> <td>3</td> <td>5217</td> <td>19</td> <td>0</td> <td>424</td> <td>215</td> <td>1</td> <td>0</td> <td>5876</td> </tr> <tr> <td>4</td> <td>633</td> <td>3</td> <td>0</td> <td>263</td> <td>30</td> <td>0</td> <td>0</td> <td>929</td> </tr> <tr> <td>5</td> <td>3596</td> <td>386</td> <td>0</td> <td>197</td> <td>4562</td> <td>0</td> <td>0</td> <td>8741</td> </tr> <tr> <td>6</td> <td>18388</td> <td>3</td> <td>0</td> <td>20</td> <td>5</td> <td>0</td> <td>0</td> <td>18416</td> </tr> <tr> <td>7</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>185</td> <td>185</td> </tr> <tr> <td>Total</td> <td>27863</td> <td>464</td> <td>0</td> <td>1355</td> <td>5393</td> <td>1</td> <td>186</td> <td>35262</td> </tr> <tr> <td></td> <td>P Total</td> <td>28327</td> <td>28327</td> <td>S Total</td> <td>6748</td> <td>6749</td> <td></td> <td></td> </tr> </tbody> </table>										Level	Clean	Dirty	Error	CLEAN	DIRTY	ERROR	Lost	Total	1	12	53	0	343	545	0	1	954	2	17	0	0	108	36	0	0	161	3	5217	19	0	424	215	1	0	5876	4	633	3	0	263	30	0	0	929	5	3596	386	0	197	4562	0	0	8741	6	18388	3	0	20	5	0	0	18416	7	0	0	0	0	0	0	185	185	Total	27863	464	0	1355	5393	1	186	35262		P Total	28327	28327	S Total	6748	6749		
Level	Clean	Dirty	Error	CLEAN	DIRTY	ERROR	Lost	Total																																																																																											
1	12	53	0	343	545	0	1	954																																																																																											
2	17	0	0	108	36	0	0	161																																																																																											
3	5217	19	0	424	215	1	0	5876																																																																																											
4	633	3	0	263	30	0	0	929																																																																																											
5	3596	386	0	197	4562	0	0	8741																																																																																											
6	18388	3	0	20	5	0	0	18416																																																																																											
7	0	0	0	0	0	0	185	185																																																																																											
Total	27863	464	0	1355	5393	1	186	35262																																																																																											
	P Total	28327	28327	S Total	6748	6749																																																																																													
Last update: Tue Jul 25 12:38:49 2006																																																																																																			

In the above example, we can see that 3 users are on the system: datafax, samer and david. datafax is using the DataFax Administration Tool, samer is using the Status Tool, and david is using **iDataFax**. In **iDataFax**, david is holding record locks. There are currently 0 new records being validated to the study database, 0 new records waiting to be validated, and 6 pending records in **iDataFax** that are awaiting completed entry. From the Database Status table we see that there are a total of 35262 records in the database. We can see that slightly more than half of the records are at validation level 6, while the other records are spread across other validation levels. We can also see that there are 186 records which have been recorded as Lost (using the Lost Data Log (**DFIdl**)), and that there are a total of 6749 (S Total) secondary records (i.e. old versions of CRFs replaced by a refax to correct some omission or error). Also we see that of the 28327 primary records in the data base, most of them, i.e. 27863 (98%) are Clean, meaning that there are no outstanding data clarification requests (QC notes) on these records.

The primary use of **DFstatus** is to check for new CRFs faxed from the clinical centers, to determine how many CRFs are currently at a specified status and validation level, and to see what DataFax tasks are currently in progress, by other users on the network. It can thus be used to determine how much work needs to be done and to determine if other users are also working on the study and what activities they have underway. For some tasks you may need exclusive access to the database or may need to coordinate your activities with other members of the data management team.

The **DFstatus** window is divided into 3 sections:

1. Server Status
2. User Status
3. Database Status

Each section can be selectively activated by clicking on the check box that appears to the left of the section name. In daily use, you will probably want to leave Server Status and User Status checked on and selectively check Database Status as the need arises.

4.2. Server Status

Most DataFax installations will have several workstations networked together so that data management staff can work simultaneously on all DataFax studies. DataFax will automatically distribute the work load across the computers on the network following a priority list installed by the DataFax system administrator. At the top of the **DFstatus** window you will see the name of the computer(s) which are currently running the DataFax Master program and the Study Server. The Master is only licensed for one machine on the network but the study server may execute on different machines depending on which machines have been authorized to run the study server, and depending whether these machines are currently running other study database servers.

This will sometimes be of interest to the system administrator but is rarely of interest to data management staff. However, if you notice that performance is very slow, you might check to see where your study server is running and then check to see if anyone is doing something else big, like running a long SAS job, on the same machine.

Additionally, the Server Status window indicates how many new records have been received by the study server and processed but not yet validated into the study database. Counts of new records are given for two mutually-exclusive criteria: those new records that are currently being validated and those that are still awaiting validation. Independently, the number of new records for which initial data entry has been started, but not yet completed, is reported as *pending iDataFax entry*.

4.3. User Status

Under User Status is a scrolling window which displays the names of all users who are currently working on the study. Beside each user name is:

- the name of the workstation or display that is running the DataFax tool they are using,
- the name of the DataFax tool,
- when they started running it,
- and whether or not the user holds locks on database records

This is helpful when a task you wish to perform depends on the completion of other tasks. For example, the person responsible for generating and sending quality control reports will want those reports to be as up to date as possible. If other users are still validating newly arrived records, the creation of new QC reports should be delayed until they are finished.

4.4. Database Status

DataFax provides a mechanism for CRF work flow management, i.e. for establishing validation or review levels for study CRFs and making sure that all CRFs pass through the specified validation steps. DataFax tracks the performance of data management tasks by stamping each record with the user's working validation level when the record is signed off in **DFvalidate**. Thus by checking the validation status of the records in the database you can determine the number of records at each stage in the studies validation process.

For example, suppose that data validation has been organized as follows:

1. The study has a full time data entry clerk who validates all newly arrived faxes to validation level 1.
2. A second data clerk double checks all records at validation level 1 raising them to validation level 2.
3. Finally, a CRA, using validation level 3, reviews all adverse event reports, all medication records and all records with attached quality control notes.

The first clerk needs to know how many new records have arrived by fax from the clinical sites. This number can be determined from the sum of the Number of records awaiting validation and Number of records being validated numbers under the Server Status window. The second clerk needs to know how many records are currently sitting at validation level 1, and are thus ready for the second validation step.

In addition to showing the total number of records at each of the 7 possible validation levels, this table also shows the number of records at each status (Clean, Dirty, Error, CLEAN, DIRTY, ERROR, Lost). In our example the CRA needs to review all records with unresolved QC notes attached to them which he/she has not already reviewed. The record counts at levels 1 and 2 under the Dirty column identifies the number of records that need to be reviewed to complete this task.

The CRA also wants to see if there are any new adverse event reports which have not yet been validated to level 3. If the adverse event reports spanned 2 pages, say plates 12 and 13, the CRA would enter 12,13 in the window beside Plates and then select Update Now from the Actions menu. The table would then be updated to show the validation level and status of all plates 12 and 13 in the database. The total number of such records at validation levels 1 and 2 can then be read from the table.

DFstatus may also be of interest to a statistician who needs to perform an interim analysis during the trial. For this purpose it would be useful to know how clean the database was for the records needed for the report. If for example, the report dealt with plates 13 and 14 from visits 0, 2 to 6 and 99, the visit and plate fields would be completed as shown below, and then Update Now would be selected from the Actions menu. The database status table would then be updated for just these records.

Figure 4.2. Specific visit and plate numbers can be specified when updating the database status.

Visits:	<input type="text" value="0, 2~6, 99"/>
Plates:	<input type="text" value="13, 14"/>



Note

The Database Status table does not include plate number 501 (the quality control report plate), unless it is explicitly entered in the plate window.

All retrieval methods available for Database Status are similar to their equivalents in the Validation Tool (**DFvalidate**) (see *DataFax User Guide, By Data Fields*).

4.5. File Menu

The File has 3 sub-menus: > Save As, Print and Exit.

- **Save As.** The Database Status report displayed in the lower half of the window can be saved to a file on disk by selecting Save As from the File menu, and completing the standard DataFax file selection dialog.
- **Print.** The Database Status report displayed in the lower half of the window can be printed by selecting Print from the File menu, and completing the standard DataFax print dialog.
- **Exit.** Selecting Exit allows you to quit the study's Status Tool but keep the DataFax Study Toolbox open.

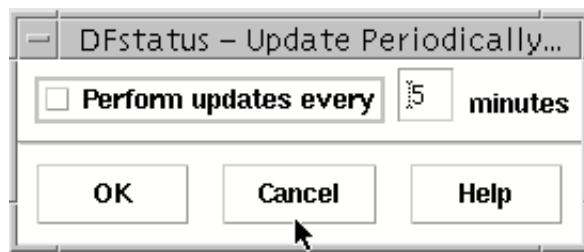
4.6. Actions Menu

The Actions is used for updating the 3 parts of the Status Tool and consists of 2 sub-menus: > Update Now and Update Periodically.

- **Update Now.** Select Update Now to immediately update the Status information in each of the 3 checked status sections. Updates will not be performed for those sections that are unchecked.
- **Update Periodically.** Occasionally you might want to leave **DFstatus** open and have it update itself automatically. This would be useful for example, if data entry staff were still validating new records and you wanted to wait until they were finished before creating QC reports.

Select Update Periodically to specify the desired interval for automatic updating. Specify the interval between updates in minutes, check on the toggle button, and then click OK. To subsequently stop periodic updates, display the dialog, check off the toggle button, and click OK as shown below.

Figure 4.3. The Update Periodically dialog box as it appears when stopping periodic updates.



Chapter 5. Validation Tool

5.1. Introduction

The Validation Tool (**DFvalidate**) displays each CRF page received by fax and its corresponding data record, simultaneously, in split screen scrolling windows. It is used to complete data entry for all newly arrived faxes, perform subsequent quality control reviews, flag problems on the CRFs with quality control (QC) notes (for internal use or transmission back to the clinical sites), and to update the database when corrections are received.

This chapter has 3 parts. We begin with a review of some basic concepts related to data validation. Then, beginning with Retrieving Data Records we describe how data records are assembled and validated. Finally, starting at Validation Tool Menus we describe each of the functions located under each of the **DFvalidate** menus.

5.1.1. Validation of New Records

Each CRF page received by fax is read by the ICR software to create an initial data record which is assigned a validation level of zero and a status of `new`, and then stored in a new record queue in the study data directory. The data record does not enter the study database until it has been reviewed and validated to a higher validation level. This step is typically performed by a data entry clerk working at validation level 1. The validation of new records involves:

- Retrieving a set of new records.
- Displaying them in the split screen windows (data above and fax below).
- Comparing what is in the data entry window with what is on the faxed image of the CRF.
- Editing the data window to enter text, completing any fields left blank or misread by the ICR software.
- Flagging any problems (e.g. missing data) with QC notes.

When the last data field has been validated the CRF page is signed off by setting its status to `clean` if there were no problems, `dirty` if QC notes had to be added, or `error` if it was not possible to read one or more of the 3 primary keys which appear on every page of a DataFax CRF (patient ID, plate, and sequence number). When status is set, the record is automatically raised to the user's working validation level and is stamped with the current date and time from the computer clock.

5.1.2. Validation of Old Records

In most clinical trials standard operating procedures will call for additional quality control reviews after initial data entry. Typically a second data entry clerk will retrieve all records at validation level 1 and review them a second time looking for any errors made during the initial data entry. When the review has been completed each record is signed off, raising its validation level, to 2 for example. Beyond this point a data manager or CRA might have responsibility for validating critical data (e.g. adverse event and patient termination reports) to higher validation levels. In addition, all records with new QC notes added by the data entry clerks, might be checked and validated to a higher level, before they are transmitted to the clinical sites in a QC report.

Changes to certain data values that were previously entered may require a reason for the data change. Which data values require this reason is dependent upon the reason level of each variable as defined in the setup, and also the current validation of the old record.

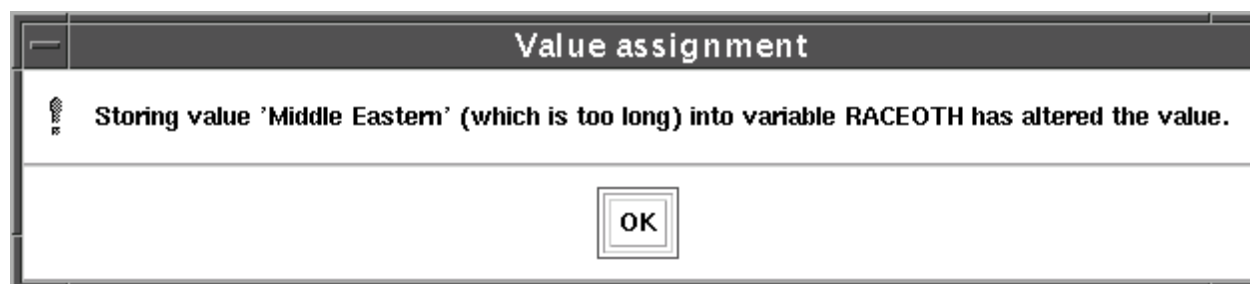
5.1.3. Validation of Old Records Following a Setup Change

Validation of existing records may also be required following a change to field formatting in the **DFsetup** tool. When the affected records are retrieved in **DFvalidate** following the change, **DFvalidate** checks the length of each existing data value against its field format. The length of each field is also checked to make sure that it is not too wide for display in the corresponding data field widget of the data entry screen. If the value length is greater than the defined maximum in **DFsetup**, the value will be shortened to the maximum based on its field type:

- String values - field value is truncated
- Date values - the field value will be set to blank
- Integer values - leading zeros will be removed if present, otherwise the field value will be set to blank
- Fixed point number values - the fractional part will be rounded to the specified digits of precision. If the result is still too wide, leading zeros will be removed if present, otherwise the field value will be set to blank

Any existing data values that are to be shortened cause a warning dialog to be displayed during record retrieval in **DFvalidate**.

Figure 5.1. The Variable assignment dialog box is invoked when a field's value is to be shortened.



You will be presented with a warning for each field in which value shortening is needed, as soon as the first record in the retrieval set becomes active. Each warning message includes the original data value, a brief explanation of the inconsistency, and the field's unique name (as defined in the database schema). The warning message is displayed in a blocking dialog which means that the OK button must be selected before continuing with the record retrieval operation. Shortened values only get inserted into the database upon record sign off. If you do not choose to sign off the record, the values will be removed for those fields in which inconsistencies exist. You will not be presented with a warning dialog if no field values require shortening.

5.1.4. Quality Control Notes

It is not uncommon to find problems on the study CRFs received from the clinical sites. A required data field may have been left blank, a recorded value may look impossibly high or low or appear inconsistent with other data recorded on the form, something written may be illegible or require further clarification, and occasionally lines may be dropped during fax transmission making a data field difficult or impossible to read.

In all of these cases it is necessary to ask the participating investigator for clarification or to correct the original CRF and refax the page. This is accomplished by attaching a QC note to problem data fields. QC notes are coded by problem type (Missing, Illegal, Inconsistent, Illegible, Fax Noise Other) and can include a query if the problem needs to be explained. QC notes may either request that the CRF page be corrected and refaxed or simply ask that a question be answered on the QC report itself and faxed back. QC notes may also be flagged for internal use only, in which case they are not included

on the QC reports faxed to the clinical sites.

All QC notes are stamped with the user's login name and the creation time and date and are then written to the QC database. They can subsequently be modified and reissued on reports as often as required. When the problem is fixed in the database the QC note is also coded as resolved in one of three ways: *resolved*, *corrected*, *resolved, not available*, or *resolved, not relevant*. Resolved QC notes remain in the QC database. This provides an audit trail of all queries issued and their eventual resolution.

5.1.5. Reasons for Data (Change)

Typically a data value is changed in a database variable because a refaxed CRF has been received that contains the data correction. In such cases, the reason for change is obvious. However, there may be other circumstances that lead to a data change which are not obvious. For example, the data entry clerk may discuss with the site a particular CRF during a telephone conversation and as a result of the conversation make a change to a data value. In such a case, the data value change should be supported with a reason.

During database setup, some or all variables may be flagged with a reason level. This defines the minimum record validation level after which a reason is required if the value in the field changes. During validation, changes to the values in such data fields will then require a reason if the record's validation level is at or above the field's reason level. In addition, once a change is made to a data field that requires a reason, it will thereafter always require a reason for subsequent data value changes, independent of what the record's validation level is. This remains true even if the data field's reason level is reset to *None* in the study setup.

The user is prompted to supply the reason (as free text, from a lookup table, or from edit check execution) at the time that the data value is changed. Reason text is entered into the Reason field of the Reason - Add dialog. The supplied reason may not contain invalid or white space only characters. If the user does not supply a reason, **DFvalidate** will provide one automatically - that is, it is not possible for the user to make a change without a reason being recorded. Alternatively, the user can choose to revert to the previous value in which case a reason is no longer needed.

Figure 5.2. The Reason dialog.

Reason - Add	
Variable	Age
Old Value	50 [reason: Clarification added by EC cr33_t3.]
New Value	51
Reason	...
<div> <div>OK</div> <div>Revert to Old</div> <div>Cancel</div> </div>	

It is also possible to add a reason for data value to any field, even though the value has not changed. The reason, in this case, may provide additional information about an unusual or illegal recorded value. To provide a reason for data value, the user must explicitly choose to add the reason because **DFvalidate** will not know to automatically prompt for one.

5.1.6. Duplicate Faxes

Some pages of the CRF may be received by DataFax more than once, either through unintentional refaxing of the same form or because of a problem flagged for refaxing with a QC note. When new pages are being validated they are checked against the existing database and any duplicates are brought to the data clerk's attention for resolution.

At this point the data clerk can examine the old version(s) of the page and compare them to the newly arrived version. Usually the new version will contain some correction and will consequently be marked as the new primary version of that page. In addition the current data record and any QC notes can be brought forward for correction and resolution respectively. If necessary, the QC notes can be modified and reissued. The new primary record will then be signed off (*clean* or *dirty*) which will reset its validation level to that of the data entry clerk (usually 1). Refaxed CRF pages will thus proceed again through all subsequent validation levels.

Only one CRF image and data record can be marked as the primary version, but all others can be retained as secondary versions. Alternatively, old versions can be deleted.

5.1.7. Intelligent Character Recognition (ICR)

The ICR software only reads check boxes, numerical fields, date fields, and visual analog scales. Text must be entered manually. Data fields will be left blank if ICR encounters an ambiguous entry (e.g. more than one response checked where only one is allowed) or if it reads a numerical value which does not pass its legal values test. The ICR software will also make errors. Accuracy of the DataFax ICR algorithms was determined using 2 independent sets of hand-written digits extracted from the CRFs faxed during a multicenter trial involving over 250 sites from several countries. The total error rate (false positives and false negatives) was 3.3% in the first set of 112,308 digits and 3.4% in the second set of 67,728 digits. The accuracy with pre-printed numbers using 18 point Avant Garde-Book font is very good with less than 1 error per 1000 digits.

Despite its limitations, most data entry clerks have found ICR to be helpful and prefer to have it turned on. However, if desired, ICR can be turned off for any or all plates in the study through the Plate > Edit in **DFsetup**.

5.1.8. Validation Levels

Each data record starts with a validation level of zero when it is created by the ICR software. Subsequent quality control reviews are each assigned a validation level in the range 1 to 7, inclusive. Data records are stamped with the user's working validation level when they are signed off.

For each study the data management coordinator must establish the desired validation steps, assign each step a validation level, and assign the various validation tasks to specific members of the study management team. Each person will have a maximum validation level at which they can work which is defined by the DataFax system administrator when the study is configured. However, users can work at any task below their maximum level by setting the validation level when the retrieval is performed. Thus, for example, a CRA with a maximum validation level of 7, could take on the role of the second data clerk to perform the second validation review by setting the working validation level to 2 when performing the retrieval.

Thus, remember to set the validation level for your retrieval to correspond to the task you are performing, and to sign off each record that you review to raise it to your working validation level. This provides a mechanism for achieving automated work flow management within the data management office.

5.1.9. Record Status Levels

Each data record will have one of the following 8 status names. Record status is set in one of 3 ways.

1. **All initial data records created by the ICR software are marked `new`.** This takes place even before the user creates a retrieval set of records.
2. **All data which are known to be permanently lost are recorded in the Lost Data Log (DFIdl).** For example, a patient refused a test or missed a visit. **DFIdl** is described in Lost Data Log Tool.
3. **By DFvalidate.** All other statuses are set by **DFvalidate** when records are signed off.

The 8 possible status names are:

- **new.** Initial data records created by the ICR software.
- **clean.** Primary record, with no problems. Data entry is complete.
- **dirty.** Primary record, with one or more problems, usually flagged by QC notes.
- **error.** Primary record, but one or more keys (patient, plate, visit) is unreadable or blank.



Note

A user must assign the status `error` to any record having one or more of its key fields blank, regardless of the user's validation level. A status of `error` will not prevent DataFax from complaining that a plate is missing, thus blank pages (i.e., with no key fields) that have been faxed into the database will be flagged as missing when the QC database is updated. Every effort should be made to determine the value of missing key fields in order that the record can at least be assigned a status `dirty`.

- **CLEAN.** Secondary data record of status `clean`.
- **DIRTY.** Secondary data record of status `dirty`.
- **ERROR.** Secondary data record of status `error`.
- **lost.** Data records recorded as permanently lost through **DFIdl**.

5.1.10. Retrieval Sets and Record Locking

DFvalidate permits the creation of multiple retrieval sets within the same session (although of course only one retrieval set can be displayed at a time). You can switch between retrieval sets at any time using Goto under the Set menu.



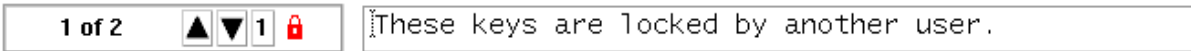
Warning

All unsaved changes will be lost when switching sets using the Set > Goto feature.

DFvalidate uses both set-based and record-based locking:

- Records in New Data Entry sets are locked as a set when the set is built and unlocked as a set when the set is released.
- Records in all other set types are not locked when the set is built - instead they are retrieved unlocked, and then locked as needed - only the current record is locked while it is displayed on-screen. This allows multiple users to have the same record in their retrieval set but only one user can validate or edit any individual record at a time.

When the user selects a record from the set, or traverses to a new record, **DFvalidate** requests the lock for that record (and all of its secondaries) from the database server. If the lock is available, it is acquired and the user can validate or edit the record. The lock is released when the user leaves the record or releases the set. If the lock is not available, the record cannot be edited, the data entry window is made insensitive and the help message area shows the following:



The user should traverse to another record, returning to this record at a later time.

This guarantees that only one user can validate or edit a record at a time.

5.1.11. Retrieval Modes

A retrieval set may be created with one of three modes:

1. **View.** Use `View` mode if you want to examine records but not modify them.
2. **Edit.** Use `Edit` mode if you want to be able to modify data records or QC notes without changing any record's validation level. This may be useful to a CRA who wants to review, and if necessary edit, all QC notes added since the last QC report was created (before sending out a new QC report), but does not want to promote data records beyond their current validation level.
3. **Validate.** Use `Validate` mode to edit data fields and QC notes, and change the status and validation level of data records. In `Validate` mode you will, by default, only be able to retrieve records which are at or below your maximum validation level. This encourages users to maintain the progression of records through increasing validation levels and prevents users from being able to change records which have passed their stage of the validation process. You can, however, bypass this feature by altering the default.

5.1.12. Signing Off Data Records

When you have finished entering or reviewing data on a record, sign it off by setting its status to `clean`, `dirty` or `error`. Signing off a data record must be done to save changes at all validation levels (not just initial data entry), and to commit the record to the database. A detailed description of signing off data records can be found later in this chapter at Signing Off Records: Setting Status and Validation Level.

5.1.13. Color Coding

DFvalidate uses color to show the status of data fields. The color coding is as follows:

`black`=background (data and fax windows)

`light green`=foreground text (data and fax windows)

`yellow`=blank, but optional, data field

`white`=legal value in data field

`red`=illegal value or missing, required/essential value in data field

`orange`=data field with pending review query or reason

`blue`=data field with unresolved query

green=data field with resolved query, approved reason, or missing value code

5.2. Retrieving Data Records

To obtain a set of records for validation, editing, or review, choose Build from the Set menu. This brings up the following dialog box for your record retrieval specifications.

Figure 5.3. The Build dialog box is invoked from the Set menu.

DFvalidate - Build ...

Retrieval Task:

Name:

Retrieval Mode: Level:

Retrieval Type:

Plate:

Maximum #:

Use	Fax Name	Id	Visit	Plt
-----	----------	----	-------	-----

The retrieval dialog has three windows:

1. **Retrieval Task window.** The retrieval task window allows a user to name and save retrieval settings for subsequent replay. A complete description of this feature can be found later in this chapter in Retrieval Tasks.

2. **Retrieval Specification window.** Here retrieval modes such as `View`, `Edit` and `Validate` can be selected along with validation level and retrieval type. A description of retrieval types begins with the section `New Records`.
3. **Retrieval Criteria window.** This window changes its contents based upon the current retrieval type.

5.2.1. Building a Retrieval Specification

A complete retrieval specification involves four steps.

1. Choose a Retrieval Mode

`View` allows you to examine records but not change them, `Edit` allows you to change data fields without changing the current validation level of the records, and `Validate` allows you to edit data fields, and change the record status and validation level. Be sure to select the appropriate mode for your task as there is no method for changing mode once a set has been retrieved (other than releasing the set, choosing a different mode, and then retrieving the set again).

2. Choose a Validation Level

Use the validation level which corresponds to the task you are performing. By default, only records which have a validation level less than or equal to the level you choose are retrieved. This can be changed by specifying a range of validation levels in the `By Data Fields` retrieval type.

3. Choose a Retrieval Type

Choose a correct retrieval type from one of the seven types available. Choosing a particular retrieval type causes the retrieval criteria window to change its contents based upon your selection. Complete the specification in the retrieval criteria window. Each of the seven windows is described separately beginning with `New Records`.

4. Select OK, Apply, or Cancel

Finally, select OK to proceed with the retrieval and dismiss the Build dialog, Apply to proceed with the retrieval leaving the dialog visible, or Cancel to dismiss the dialog without executing the retrieval.

5.2.2. New Records

Retrieval Type:


 A screenshot of a software interface showing a dropdown menu for 'Retrieval Type'. The text 'Retrieval Type:' is on the left. To its right is a rectangular box containing the text 'New Records' and a small square icon on the right side, indicating a dropdown menu.

New records are created by the ICR software when the computer receives a study CRF and are inserted into the new record queue.

The process of building a retrieval set from the new record queue requires two steps:

1. Pre-select new records from the queue by plate number and/or by maximum number of new records.

The default criteria is to select new records from all plates up to a maximum of 4096 new records. If you know in advance that you will only want to validate certain plate numbers or a certain maximum number of records, you can speed up the retrieval process by specifying those criteria first.

Figure 5.4. Pre-selection criteria that restrict the new record retrieval to the first 100 records having plate numbers in the range of 1 to 7.

Retrieval Mode:

Validate ☐

Level:

1 ☐

Retrieval Type:

New Records ☐

Plate:

...

1-7

Maximum #:

10

Retrieve

Use

Fax Name

Id

Visit

Plt

After you have specified the pre-selection criteria, or simply used the defaults, click the Retrieve button. This causes **DFvalidate** to request the new records from the new record queue. The resulting pre-selected records are displayed in the scrolling list.

Figure 5.5. The scrolling list displays all pre-selected records.

Retrieval Mode: Level:

Retrieval Type:

Select Records

0 selected of 10 total

Use	Fax Name	Id	Visit	Plt
	9807/0048002	33002	0	1
	9807/0048003	33002	1	2
	9807/0048004	33002	0	7
	9807/0048005	33002	2	4
	9807/0048006	33002	21	5
	9807/0048007	33002	22	5
	9807/0048008	33002	23	5
	9807/0048009	33002	24	5
	9807/0048010	33002	0	6
	9807/0048011	33002	0	7

2. Choose a retrieval set of records from the scrolling list.

The scrolling list shows the name of each fax image (CRF page) and the key fields (patient ID number, visit, and plate) as read from it by the ICR software. Each fax image name has the format *yyww/ffffppp*, where *yyww* is the year, and week of the year, in which the fax was received, *ffff* is the sequential number of the fax received that week, and *ppp* is the page number within that fax. Records that appear in the list with the phrase *In Use* to their left, are records from the new record queue that have already been selected and locked by another user. You will not be able to select them.

To choose records from the list, you can use the mouse to highlight individual records or groups of records. As you highlight records that were not previously chosen, the word *Selected* appears next to those records. The total number of records selected is incremented and displayed in a counter next to the *All* button. If you highlight records that were previously chosen (and had the word *Selected* next to them), those records are no longer chosen and the number of records selected is decremented appropriately.

Alternatively, you can choose records from the list by using the selectors at the top of the window.

It is best to retrieve an entire fax at a time. This can be done by entering the fax number (*yyww/ffff*) into the field to the right of *Fax Name* and then clicking on *Fax Name*. Any records in the list that have a matching fax name are chosen and marked *Selected*.

**Note**

Choosing records that were previously selected does not de-select those records as would occur using mouse-based selection.

You can also specify wild-card characters in the fax number; for example **9807/004?** would match all faxnames from **9807/0040** through **9807/0049**. The example shows the selected records after a fax name of **9807/004800?** was entered and Fax Name was clicked.

Figure 5.6. The fax name selection criteria 9807/004800? selects a total of 8 records.

The screenshot shows a software window for record selection. At the top, 'Retrieval Mode' is set to 'Validate' and 'Level' is '1'. 'Retrieval Type' is set to 'New Records'. Under 'Select Records', 'By Fax Name' is selected with the input '9807/004800?'. Other options like 'By Plate', 'Maximum #', 'All', and 'None' are present but not selected. A status message on the right says '8 selected of 10 total'. Below is a table of records.

Use	Fax Name	Id	Visit	Plt
Selected	9807/0048002	33002	0	1
Selected	9807/0048003	33002	1	2
Selected	9807/0048004	33002	0	7
Selected	9807/0048005	33002	2	4
Selected	9807/0048006	33002	21	5
Selected	9807/0048007	33002	22	5
Selected	9807/0048008	33002	23	5
Selected	9807/0048009	33002	24	5
	9807/0048010	33002	0	6
	9807/0048011	33002	0	7

Records can also be selected by plate number. Enter the plate number into the field next to By Plate and then click on By Plate. Any records in the list that have a matching plate number, and are not already selected, are chosen and marked as Selected.

Clicking All selects all of the available records (those not marked In Use), while clicking None resets the list by de-selecting and previously selected records.

Remember to finish your selection by clicking OK or Apply from the control area at the bottom of the window. This completes the process of selecting records from the new record queue.

**Important**

You should make your selection from the new record queue as quickly as possible because as long as you have the window visible other users are prevented from getting new records.

5.2.3. *By Data Fields*

Retrieval Type:**By Data Fields**

Records which have been validated to at least level 1 are written to the study database and can be retrieved by a combination of:

- Center number
- Key fields (Patient ID, Visit Number and/or Plate Number)
- Validation level
- Current record status (Clean, Dirty, etc.)
- Creation and modification dates
- A string for a pattern search.

When multiple specifications are made only records which meet all of the specifications will be retrieved.

The By Data Fields retrieval window has the following appearance.

Figure 5.7. The By Data Fields retrieval window allows retrieval of data which has been validated and committed to the database.

Center:	<input type="text"/>
Patient Id:	<input type="text"/>
Visit:	<input type="text"/>
Plate:	<input type="text"/>
Validation:	<input type="text"/>
Creation Date:	<input type="text"/>
Modification Date:	<input type="text"/>
Pattern:	<input type="text"/>
Status:	<input checked="" type="checkbox"/> clean <input checked="" type="checkbox"/> dirty <input type="checkbox"/> error <input type="checkbox"/> CLEAN <input type="checkbox"/> DIRTY <input type="checkbox"/> ERROR <input type="checkbox"/> lost <input type="checkbox"/> delete <input type="checkbox"/> all

Each of the Center, Patient Id, Visit, Plate, Validation, Create Date, and Modification Date fields will accept any combination of single value, range of values, list of values, or list of range of values.

There are a few points that you need to keep in mind when specifying your retrieval criteria.

1. **It is not possible to specify both Center and Patient Id selection criteria in the same retrieval.** If both fields contain values, an error message is generated and the retrieval is cancelled.
2. **If Center specification includes:**
 - one or more illegal center numbers, an error message is displayed and the retrieval is aborted. An illegal center number is one that falls outside the system-defined range of 0-2146.
 - one or more unknown center numbers, the retrieval proceeds but the unknown center numbers are ignored. An unknown center number is one that falls within the system-defined range of 0-2146, but does not match any of the center numbers in the centers database.
3. **Date values must be specified in yy/mm/dd format.** yy/mm/dd is the DataFax default format for dates.
4. **The symbol to indicate a range can be either - or ~.** For example, both a range of 99001-99020 and 99001~99020 would be valid Patient Id specifications.
5. **The symbol to delimit items in a list is ,.** Both 99001, 99005 and 99001,99005 are valid Patient Id specifications.
6. **Retrieving by Modification Date will not give you records that simply had QC notes added.** Adding a QC note does not change the modification date and time stamp of a record unless the record is assigned a different status or there are changes to a data field(s).

To retrieve records that simply had QC notes added, go to the QC Tool (**DFqc**) and retrieve records by QC note creation date, save the information as a DataFax Retrieval File (DRF), then retrieve this DRF file in **DFvalidate**.

7. **Retrieving by Modification Date will not give you records that simply had their validation level changed.** As for QC notes, changing the validation level of record does not change the date and time stamp of that record unless there have been changes to the record status or a data field(s).
8. **For retrieval by Pattern, you must delimit the search string with the wildcard character, *.** The pattern that is searched for must also be specified in the same case, as these comparisons are always case sensitive. For example, to search for the pattern **ASA**, specify the pattern criteria as ***ASA***.
9. **If you specify the same value more than once in a retrieval list, DFvalidate will try to retrieve records using the same criteria twice.** This will lead to a message indicating that a certain number of records could not be retrieved because they are already in use.

Each of the following is a valid retrieval specification.

Figure 5.8. Examples of valid retrieval specifications in the By Data Fields retrieval window.

Patient Id:

Visit:

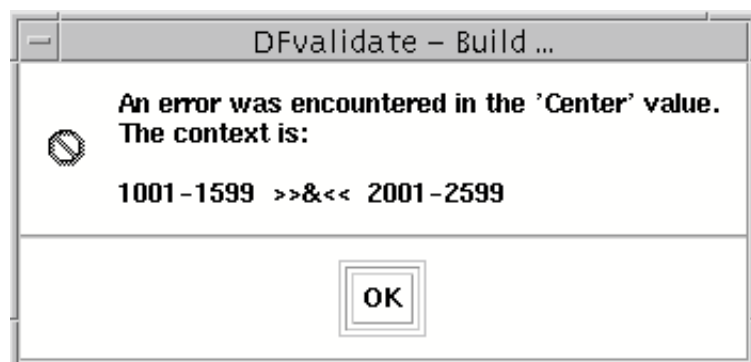
Validation:

Creation Date:

If you make an error in a retrieval specification, there are two likely outcome scenarios.

1. The set of records that you retrieve, if any, will not be the set that you intended with your criteria. The validation tool does not check the semantics of your retrieval specification.
2. If your specification contains a syntax error or an illegal value, the following error dialog will appear.

Example 5.1. An error message dialog appears when DataFax encounters a syntax error in retrieval specifications.



In this case, **&** is not a valid list separator.

Finally, remember to select the record statuses that you would like to restrict the retrieval to. The default is to retrieve only **clean** and **dirty** primary records. If you select **status all**, that is equivalent to selecting each of the other record statuses.

Carefully review your retrieval criteria, and remember to finish your selection by clicking OK or Apply from the control area at the bottom of the window.

5.2.4. By DataFax Retrieval File

Retrieval Type: **By DataFax Retrieval File** ☐

A DataFax Retrieval File (DRF) is an ASCII file in which each record identifies the patient ID number, sequence (or visit) number, CRF plate number (in that order, | delimited) and optionally the image ID corresponding to a record in the study database.

This retrieval method requires the name of a DataFax Retrieval File as its criteria. If a relative file name is specified, DataFax assumes that the file is in the study `work` directory. If the DataFax Retrieval File is located anywhere else you must specify the full path name of the file. In the following example, the DataFax Retrieval File `VDillegal.drf` is expected to be in the study `work` directory.

Figure 5.9. The record retrieval dialog with retrieval By DataFax Retrieval File specified.

DFvalidate - Build ...

Retrieval Task:

Name:

Retrieval Mode: Level:

Retrieval Type:

Directory Filter: *.drf

File	Description	Creation
VDillegal.drf	Illegal visit dates	valid1 99/04/22

File

Description

There are a few important points to remember when using DataFax Retrieval Files:

- Records will be assembled in the retrieval set in the same order in which they appear in the file.
- Any non-unique keys will be removed prior to the building of the retrieval set.
- The current validation level of the retrieved records is not tested against the retrieval validation level. All records from the DataFax Retrieval File, regardless of validation level, are retrieved.

When records are retrieved by DRF, the DRF filename is displayed in the Record List window of the Validation tool directly below the Image ID information.

The retrieve by DRF facility allows **DFvalidate** to work in conjunction with special purpose programs which identify problem cases in need of review. For example the study programmer might write a special integrity check program which looks for inconsistencies and writes a report of any problems identified along with a corresponding DRF. A data manager or CRA could then periodically run this program, and with the report output in hand, could retrieve all of the records referenced in the corresponding DRF to fix problems or add QC notes.

5.2.5. By Program

Retrieval Type:

By Program ☐

This retrieval method is not unlike the previous one. The program name given as the criteria must output records formatted exactly like those in a DataFax retrieval file. This is useful in cases where the program does not create a printed report but simply produces a retrieval set.

This retrieval method requires a UNIX command as its criteria. You can specify any legal UNIX command line, including argument lists. The requirement is that execution of the program produces properly formatted keys written to standard output.

If a relative name is specified, DataFax assumes that the file (or program) is in the study **DFreports** directory. If a file (or program) is located anywhere else you must specify the full path name of the file.

Figure 5.10. The By Program retrieval options uses DFget to extract the patient ID, visit number, and plate number from each record in plt001.dat.

Program:

DATAFAX_DIR/bin/DFget 7 6 5 ../data/plt001.dat

5.2.6. By Variables

Retrieval Type:

By Variables ☐

Retrieval queries built with this retrieval type consist of logical expressions referencing variable names from the study database schema. The current implementation limits you to specifying an expression referencing variables and values from one plate at a time. When building retrieval sets by variable, the user can specify the validation level of the records which are to be retrieved. The validation level can be a single value, range of values or list of values. For example, validation levels of 1, 1-7, and 1, 3, 5 are all valid specifications.

To build an expression:

1. Set your Retrieval Mode (*Validate*, *Edit*, *View*), the validation level at which you wish to work, and the validation level at which you want to retrieve records.

The validation level at which you wish to work is set the Level toggle button.

2. Choose the plate that is of interest to you.

Select the plate from the left column. Selecting a plate updates the middle column with the variable names from that plate. Once you have selected a plate, follow steps 3 to 6 to build a query.

3. Select a variable from the middle list.

Notice that it is inserted into the Query field with some extra punctuation around it. This punctuation is required by the parser in **DFvalidate** - do not edit it. If the selected variable contains any coding (i.e., the variable is a choice or check field), codes and their respective labels will appear in the Codes window near the bottom of the variable retrieval dialog.

4. Select a conditional symbol from the right list.

Most queries involve testing a variable against a value or range of values. To do this a conditional operator is required. If the operator you need is not listed, you can simply type the operator into the Query field.

5. Indicate the conditional value from the keyboard.

If you are testing against a string value, remember to enclose the value in double quotes (").

6. Repeat from step 3 as necessary.

If you are concatenating conditions, you will need to specify a conditional operator such as '&&' or '||' to join the conditions.



Note

You can also enter the query or components of the query directly from the keyboard - the scrolling lists are there as a convenience and a memory aid.

Figure 5.11. This query demonstrates a test for males (sex == 1) with a systolic blood pressure of 200 mm Hg or higher (sbp >= 200).

DFvalidate – Build ...

Retrieval Task:

Name:

Retrieval Mode: Level:

Retrieval Type:

Query:

Validation:

Plates	Variables	Symbols
1 - Blood Pressure Scree	ID001	(
2 - Patient Entry Form	PINIT001)
3 - Medical History	AGE	== EQUALS
4 - Current Medications	SEX	!= NOT EQUAL
5 - Monthly Follow-up Fc	RACE	~ CONTAINS
6 - Patient Evaluation Re	RACEOTH	>= GREATER EQUAL
7 - Study Termination	S1DATE	> GREATER
8 - Medication Log	S1SBP1	<= LESS EQUAL
9 - Adverse Event Repor	S1DBP1	< LESS
10 - Death Report	S1SBP2	OR
20 - Date Log	S1DBP2	&& AND
501 - Quality Control Rep	S1SBP3	+ ADD
	S1DBP3	- SUBTRACT
	S2DATE	* MULTIPLY

Codes

The constructed query is translated into a UNIX awk script by **DFvalidate** and passed on to awk for execution. The result of the execution is a DataFax Retrieval File written to the standard output and subsequently read by **DFvalidate**.

5.2.6.1. Retrieval by Date Variables

Retrieval by date variables is a special case of Retrieval by Variables. In the case of date variables, all comparison operations must be done mindful of date semantics. If a comparison is required between two date variables, this can be done in the above mentioned manner, namely selecting one date variable, a condition symbol, and then another date variable. For example: `$(CONSENTDATE) == $(ENROLLDATE)` or `$(VDATE) > $(EXAMDATE)`

When a date difference is needed, it is important to remember that date arithmetic is performed in units of days. For example, to locate all records where the exam date is more than 7 days previous to the visit date, the following might be used: `$(VDATE) - $(EXAMDATE) > 7`

If a constant date value is required, the constant value must be expressed in a specific notation, and that notation is: `julian("1999/07/15")` This represents the constant value for July 15, 1999.

Comparing a date variable against a fixed date value then requires the following retrieval query: `$(VDATE) >= julian("1999/07/15")` This retrieves all of the records where the visit date is on or after July 15, 1999.



Note

The fixed date must appear as a 4 digit year, followed by a / followed by a 2 digit month, followed by a /, and finally a 2 digit day of the month. Any other representation of a fixed date will lead to unexpected results.

5.2.7. By QC Report

Retrieval Type:

By QC Report



This retrieval method allows you to assemble all old data records which are referenced in one or more quality control reports. The name(s) of the desired report(s) must be specified as the criteria. When specifying names remember the following:

- DataFax quality control report names have the format `ccc-yyymmdd`, where `ccc` is the center number, and `yyymmdd` is the year month and day the report was created. This number is printed at the top of each page of the quality control report.
- Report names must be separated by a space when entered as a list.



Important

Consider the following scenario: A QC report is created and successfully sent to a center, then a patient having QC notes in the report is moved to a different center. When this occurs, DataFax will immediately update the QC notes so that they belong to the new center number. Thus, if you attempt to retrieve the the patient's QC notes using the original report name (old center number plus report date), they will not be found. You can trick DataFax into retrieving the desired records by specifying the new center number plus report date when performing a By QC Report retrieval. But remember that this is just a trick because the QC notes were sent to the original center not to the new one; and they will appear in a printed copy of the QC report sent to the old center, not the new one. Future QC reports will of course be sent to the new center.

Figure 5.12. By QC Report is used to retrieve all of the records referenced in any QC report generated on May 4, 1995 (95/05/04) and also the records in the QC report of May 10, 1995 (95/05/10) for center 011.

QC Report(s): *-950504 011-950518

5.2.8. By Double Data Entry Batch File

Retrieval Type: By DDE Batch File ☐

When this retrieval type is selected, a scrolling list of available batch files appears. To choose a batch file from the list, simply highlight it with the mouse or use the up and down arrow keys to move the active list selection and the space bar to highlight the active selection.

Figure 5.13. Selecting By Double Data Entry Batch File will display a scrolling list of available batch files.

Batch File:

- test01.dde
- test02.dde
- test03.dde
- test04.dde**
- test05.dde
- test19.dde
- test06.dde
- test07.dde
- test08.dde
- test09.dde
- test10.dde
- test11.dde
- test12.dde
- test13.dde
- test14.dde

During double data entry, the retrieval mode is indicated as Dbl Entry in the record list window. QC notes and reasons can be added but there is no duplicate testing or duplicate resolution.

5.2.9. Raw Data Entry

Retrieval Type: Raw Data Entry ☐

Raw data entry is a method by which data may be entered into the database without corresponding CRF fax images. This retrieval method allows you to specify a set of plates and visit numbers to be entered for a particular patient(s). Once retrieved, you will see blank data records that match the retrieval criteria specified. There will be no CRF images for these data records in the lower half of the screen. Validation by Raw Data Entry may be used for entering data from paper copies of CRFs that, for whatever reason, can not be faxed into the database. Raw Data Entry can also be used for defining patient test cases during

testing of a new DataFax study setup. Complete patient test cases can be defined but only key fields entered in order to test multiple scenarios in the study visit map.

To create a set of records for raw data entry, the following steps should be performed.

1. Set Retrieval Type to Raw Data Entry.

The Raw Data Entry retrieval window has the following appearance. The Patient Id and Visit fields will accept a single value, range of values, list of values, or list of range of values. The Plate field will accept a single value or a list of values, but not ranges of values.

Figure 5.14. The Raw Data Entry retrieval dialog

DFvalidate - Build ...

Retrieval Task: **Delete**

Name: **Add**

Retrieval Mode: ☐ **Validate** **Level:** ☐ **1**

Retrieval Type: ☐ **Raw Data Entry**

Patient Id:

Visit:

Plate:

- 1 - Blood Pressure Screening Visits
- 2 - Patient Entry Form
- 3 - Medical History
- 4 - Current Medications
- 5 - Monthly Follow-up Form
- 6 - Patient Evaluation Report

Add Case ☐

or

Defined:

Add Case ☐

Cases:

Delete Case ☐

OK **Apply** **Cancel** **Clear**

2. Enter a Patient Id or list of Patient Ids and a Visit number or list of Visit numbers.

The symbol to delimit items in a list is ,. The symbol to indicate a range is - or ~.

3. From the scrolling list of Plates, select those plates that you want to include in your retrieval set.

To select a plate, simply click on its entry in the scrolling list. The plate entry will then be highlighted. To deselect an entry, click on the highlighted entry. You may select multiple plates for inclusion in a single retrieval set, however, be sure that all selected plates do in fact occur at each Visit specified in the retrieval criteria.

4. Click on the Add Case button to the left of the scrolling list of plates.

The criteria specified for Patient Id, Visit and Plate will be summarized in the entry or entries appearing in the Cases window at the bottom of the set retrieval dialog. At least one case must be added before proceeding. If no cases have been added, an error dialog informing you of this will appear.

Figure 5.15. A case consisting of Plates 2, 3, 4 for Visit 1 and Patient Id 99988 is specified for retrieval.

DFvalidate – Build ...

Retrieval Task:

Name:

Retrieval Mode:

Retrieval Type:

Patient Id:

Visit:

Plate:

- 1 - Blood Pressure Screening Visits
- 2 - Patient Entry Form
- 3 - Medical History
- 4 - Current Medications
- 5 - Monthly Follow-up Form
- 6 - Patient Evaluation Report

or

Defined:

Cases:

5. Repeat steps as necessary

The above steps must be repeated for each case that you want to include in your retrieval set.

6. Click OK to build a retrieval set of cases

Blank records for each record in the set are constructed, the first one of the set is displayed, and the Build dialog closes.

It is also possible to build a retrieval set from a pre-defined collection of cases. The pre-defined collection of cases must be done in advance following the description in *DataFax Programmer Guide*, *DFraw_map - raw data entry map*.

5.2.10. Retrieval Tasks

The process of selecting or entering each and every criteria that is required to build a specific retrieval set can be tedious, especially if it is repeated often because you typically perform the same retrieval several times a day, or several times a week.

To simplify this specification process, you can tell DataFax to remember a named collection of retrieval criteria so that you can subsequently reference and restore those retrieval criteria by their name. This named collection of retrieval criteria is called a retrieval task, and you can create as many different retrieval tasks as you require.

Retrieval tasks are automatically saved for you in the file `.DFtasks.user` in the study directory (see *DataFax Programmer Guide*, *.DFtasks.user - user tasks* for more detail). The format of the file is plain text and can be edited if you are careful, but we recommend that you not edit this file by hand, as modification of the file is straight forward through **DFvalidate** for validation tasks or **DFqc** for QC tasks.

5.2.10.1. Create a Retrieval Task

To create a new retrieval task:

1. Set the various retrieval criteria in the Build window.

This includes specifying the Retrieval Mode, the Retrieval Level, and all of the relevant settings for the chosen Retrieval Type.

In the example below, the criteria have been set for a first level data entry task.

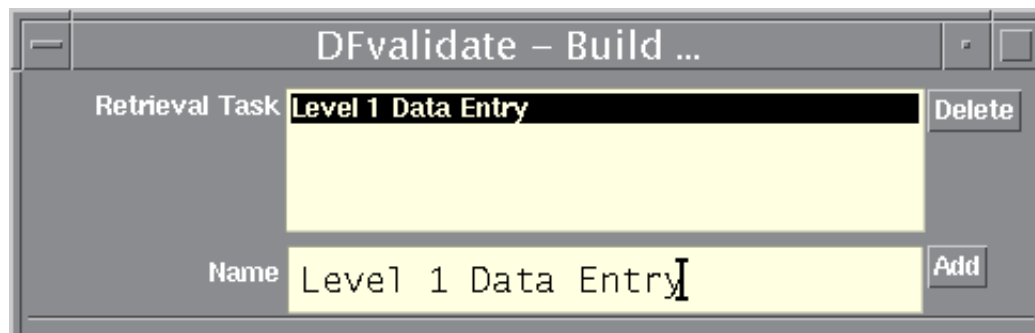
2. Enter a unique name for the retrieval task into the Name field.

In the example, we name the task **Level 1 Data Entry**.

3. Click Add.

The retrieval criteria are determined and saved in the task named **Level 1 Data Entry**.

Figure 5.16. The new task name Level 1 Data Entry is added to the scrolling list of defined tasks.



You can define as many retrieval tasks as you like. Each new task is added to the list in alphabetic sorted order.

To select a retrieval task, simply select the task name from the scrolling list. Single-click selection is all that is required.

5.2.10.2. Modify a Retrieval Task

To modify an existing task:

1. Select the task that you want to modify from the scrolling list.

Single-click selection is all that is required.

2. Make modifications to the retrieval criteria.

Multiple retrieval criteria can be modified at the same time.

3. Click Add to overwrite the existing criteria with the modifications.

If you have not changed the name of the retrieval task, it will remain the same in the scrolling list of tasks.

5.2.10.3. Delete a Retrieval Task

To delete an existing task:

1. Select the task that you want to delete from the scrolling list.

Single-click selection is all that is required.

2. Click Delete.

The name of the retrieval task will disappear from the scrolling list of task names.

Any modifications you make to your retrieval tasks are automatically updated to the `.DFtasks.user` file in the study directory.

5.3. Validating Data Records

5.3.1. *Validation Tool Windows*

When you have completed your retrieval specifications and clicked on OK, all available records which match the retrieval specifications will be assembled and displayed in 2 vertically stacked windows. This vertical window configuration is the DataFax default. The data record (as it exists in the database) is displayed in the upper window and the corresponding faxed page of the CRF appears in the lower window. Between these 2 windows is a small status window, which contains page turning buttons and a duplicate record button, and is also used to display help messages when needed. The help message window can accommodate lengthy text and is scrollable. The `Home` and `End` keyboard keys may also be used to navigate through lengthy messages provided the appropriate keyboard mapping exists in your system. Text in the help message window is non-editable.

Figure 5.17. 1 of 29 records is displayed as shown by the record page count in the middle status window.

DFvalidate [samer] Demo Study 053

File Set Record QC Reason Window Misc Help

DataFax #253 253 Plate #002 2 Visit #001 1

Patient Number 03002 Patient Initials JAL Entry Date 10/01/01
F M L day month year

PATIENT ENTRY FORM

Eligibility

Does the patient still meet the eligibility criteria? ☒ Yes ☐ No → Explain

Randomization

Provided the patient meets all eligibility criteria, call the randomization hot line ((905) 999-8888) for this patient's random medication code number.

Medication Code #: 3751

1 of 6

DFvalidate [samer] Demo Study 053

DataFax #253 253 Plate #002 2 Visit #001 1

Patient Number 03 002 Patient Initials JAL Entry Date 10 01 01
F M L day month year

PATIENT ENTRY FORM

Eligibility

Does the patient still meet the eligibility criteria? ☒ Yes ☐ No → Explain

Randomization

Provided the patient meets all eligibility criteria, call the randomization hot line ((905) 999-8888) for this patient's random medication code number.

Medication Code #: 3 7 5 1

Alternatively, the main **DFvalidate** window can be set to a horizontal configuration. This can be accomplished by selecting Horizontal from the Window. In horizontal layout, the data window is on the left, the CRF window is on the right, and the status window spans both at the bottom. As it is not possible to perfectly save and restore the size of each window as the layout is toggled between vertical and horizontal, the windows can be manually re-sized as desired.

**Note**

Window configuration can be customized for each user through the DataFax application defaults settings (see Customizing appearance and behavior). The app-default setting `DFvalidate*.viewHorizButton.set:False` is the default, resulting in a vertical window configuration upon opening **DFvalidate**. If this same setting is changed to 'True', the horizontal window configuration will be the default upon starting **DFvalidate**.

In addition to the main validation window there are five optional windows that you can selectively pop-up or pop-down. These five windows are:

- Record List
- Meta data Window
- Edit Checks Log Window
- Variable Window
- Clipboard

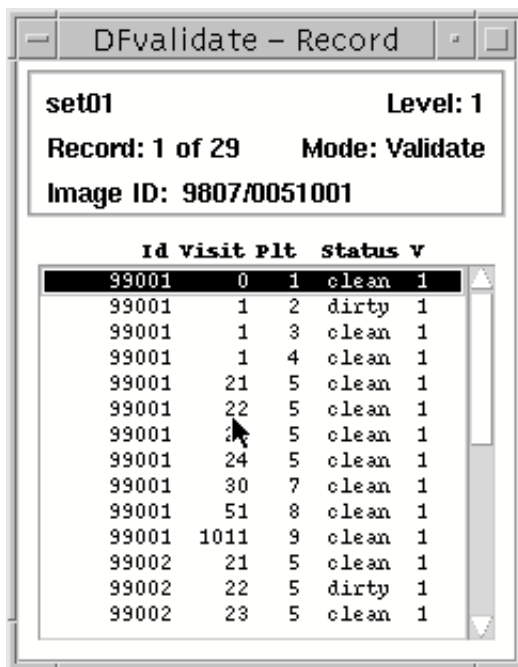
You can customize which windows appear when **DFvalidate** is started by changing your application defaults (your DataFax system administrator can help you with this), and also change which windows are currently displayed by toggling their visibility setting in the Window menu.

5.3.1.1. Record List

The record list window summarizes the retrieval settings of the current set and provides a one-line description of each retrieved record. The summary of the retrieval set appears at the top and includes the set name (the unique reference name for the set generated by **DFvalidate**), the set's validation level, the set's mode, the set size and which record of the set is currently being displayed in the main validation window.

All records in the current retrieval set are listed in a scrolling list in the bottom section of the record list window, sorted by Patient ID, Visit, and Plate number. The current record status (new, clean, dirty, etc.) and validation level (0-7) are also displayed. If you are working in `Edit` or `Validate` modes, the window contents are updated each time a change is made to a record. Any changes which you have made to its keys will be displayed in this window, along with the records new status (if you changed it). Also, if you are working in `Validate` mode, the record's changed validation level will be updated.

Figure 5.18. A record list window, with a resized scrolling list.



Id	visit	plt	Status	v
99001	0	1	clean	1
99001	1	2	dirty	1
99001	1	3	clean	1
99001	1	4	clean	1
99001	21	5	clean	1
99001	22	5	clean	1
99001	23	5	clean	1
99001	24	5	clean	1
99001	30	7	clean	1
99001	51	8	clean	1
99001	1011	9	clean	1
99002	21	5	clean	1
99002	22	5	dirty	1
99002	23	5	clean	1

5.3.1.2. Meta Data Window

The Meta Data window displays the QC note and reason information for the current variable. When the Meta Data Window is displayed, its contents are updated each time a new variable becomes current, either through normal keyboard traversal or selection with the mouse.

Figure 5.19. The Meta Data window.

Variable **Study Week**

QC

Status **New, in report**

Created **femy 06/07/07 15:00:59**

Resolved

Problem **Missing**

Use **External**

Type **Refax**

Query

Reply

Note QC_QCI

Reason

Status

Created

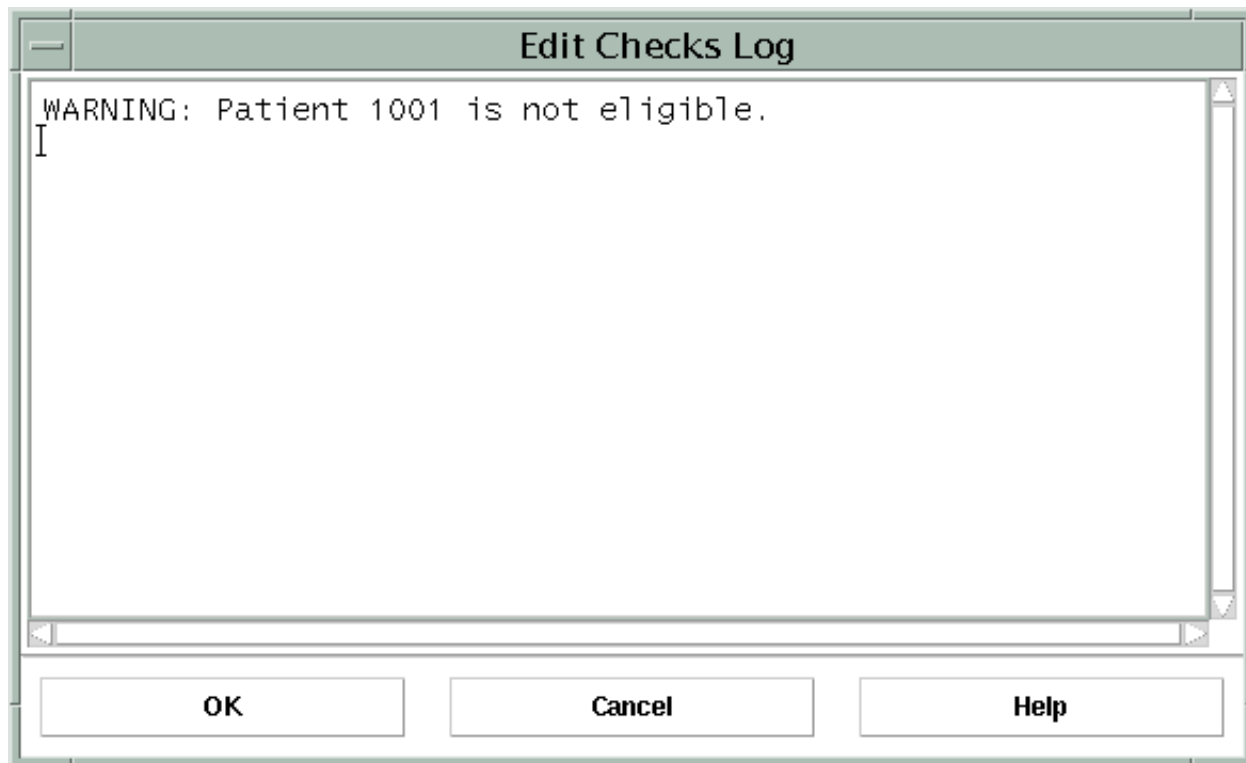
Reviewed

Reason

The fields in the Meta Data Window are not editable. If you want to make modifications to an existing QC or reason, you must choose the appropriate item from the QC or Reason menus. QC menu.

5.3.1.3. Edit Checks Log Window

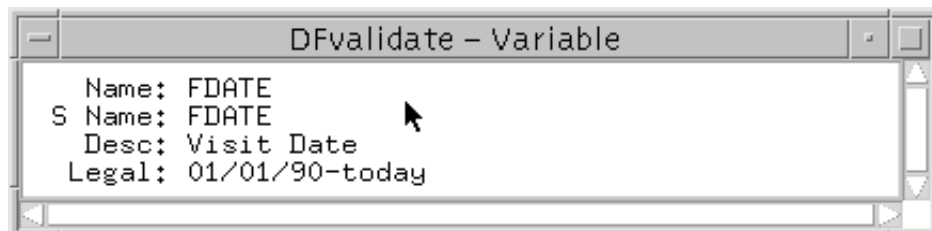
This window displays any warning and error messages that have been programmed in the edit checks for this study, and subsequently triggered during data review in **DFvalidate**. When any such message is triggered, the Edit Checks Log window will be made visible if not already so. The window has the following appearance.

Figure 5.20. The Edit Checks Log window.**Note**

The contents of this window are not saved across invocations of the validation tool.

5.3.1.4. Variable Window

The Variable Window displays the schema, or data dictionary, information for the current variable. When the Variable Window is displayed, its contents are updated each time a new variable becomes current.

Figure 5.21. The Variable window

The Variable Window reports the unique variable name, generic variable name, description, and legal value definition from the study schema.

5.3.1.5. Clipboard

The clipboard is a convenience window for copying and pasting text from one string field to another. The clipboard window is not linked to a specific data field and may be active while traversing through fields or across records.

To copy text from a string field into the clipboard:

1. Double click on the text to be copied (this highlights the text).

You can also highlight the text to be copied by dragging the mouse over it.

2. Position the cursor within the clipboard to make it active.

The cursor inside the clipboard will flash.

3. Press Shift-Insert to paste the text into the clipboard.

The inserted text will be displayed in a reduced font size, thereby making more text visible without scrolling.

Figure 5.22. The Clipboard window.



5.3.2. Moving Through the Records in a Retrieval Set

You can move through the records in a retrieval set in the several ways list below. Be careful when using these methods however because, depending on your preferences settings, you can inadvertently move to another record without signing off your changes for the current record.

- When you sign off a record, in any of the ways described in Signing Off Records: Setting Status and Validation Level, the next record in the retrieval set will be automatically displayed. If you are on the last record in the set you will be returned to the first record and a warning beep will sound.
- If you have the record list window open (always recommended) you can also move to any record in the retrieval set by selecting the line which identifies that record in the scrolling record list. Be careful when using this feature however because you can lose any unsaved changes you make to the current record when the Ask for confirmation to switch record without saving changes preference is turned off (see Preferences).
- With the mouse cursor positioned anywhere in the scrolling list of the record list window, you can move to the next record by pressing the Down arrow key, or to the previous record by holding down the Up arrow key.
- The information window that lies between the upper data window and the lower CRF window, contains a small form with page turning controls on it. You can move to the next record by pressing the down button, or move to the previous record by pressing the up button.
- When you reach the last record in your retrieval set any of the above methods of moving to the next record will move you back to the first record in the set. A beep will sound to draw your attention to this. Similarly, when the current record is the first record in the list, moving to the previous record will move you to the last record in the set.

5.3.3. Editing Data Fields

5.3.3.1. Moving Through Data Fields

Data validation involves moving through the fields in the data window, comparing each one with the contents of the fax image and making any corrections or additions that are required. The most common way to move through the fields in the data window is to use the Tab key. Each time you press the Tab key, **DFvalidate** will move to the next field. Holding down the Shift key while pressing Tab will move you backwards through the data fields, one field at a time. You can also move directly to a data field by selecting it with the mouse in the data window.

When a record is first presented, both the data and CRF windows are scrolled to the top and the first data field is made the current field. The current field is indicated in two ways depending on the field type. For numeric, date, and string fields, the current field is indicated by an additional highlight rectangle drawn just outside the rectangle enclosing the data value, and the text cursor flashes. For choice and check fields, the current field is indicated by a highlight rectangle drawn around the current choice.

5.3.3.2. Skip Field Conditions

There may be skip conditions on the page which will cause **DFvalidate** to jump over fields when some condition is met. For example, the data window might be set to skip over drug information fields if a leading question 'Is the patient taking any other medications?' is answered 'no'. If the record status is `new`, and the skip condition is true, the specified number of fields will be skipped until at least one of the following conditions is met:

- the requested number of fields has been skipped
- the current field to be skipped is not blank

The second bullet above indicates that the skip behavior will abort at the first field in which data is present, regardless of whether all the requested fields have been skipped.

If the record status is anything but `new` (the record has been previously validated), and the skip condition is true, all of the requested fields will be skipped, independent of their contents.

5.3.3.3. Modifying the Contents of the Current Field

You can move inside a data field to delete and/or insert characters by selecting the text cursor location with the mouse or with the left and right arrow keys on the keyboard. The Delete key will remove one character to the right of the text cursor, while the Backspace key will remove one character to the left of the text cursor. Text is always inserted immediately before the text cursor.

Additional methods for modifying the contents of data fields are available through the standard Motif user interface. These are described in X11/Motif and DataFax User Interface.

5.3.3.4. Scrolling Text Windows

Each numeric or text data field has a maximum storage length and a maximum display length. Often the display length is shorter than the storage length. This allows you to enter long comments without requiring that there be enough room on the screen to display it all. When you type a comment which exceeds the display length of the field, the contents of the field automatically scroll to the left. To see the entire text value, select the field to make it current and be sure that the clipboard window is visible. The text can be copied into the clipboard where it can be easily reviewed.

5.3.3.5. Lookup Tables

A lookup table of standard entries can be attached to any text field, e.g. drug names, dose descriptions, disease/symptom lists, adverse events, etc. This is done when the study database is defined. It requires 3 steps.

1. **Each lookup table must be created using an editor or program to put the lookup entries into DataFax format.** Each entry in a lookup table appears on a separate line and consists of a search string or acronym (e.g. BB), and a result string, which is the text string to be entered into the database (e.g. Beta Blocker). These 2 components are separated by the | field delimiter. Remember that DataFax uses the | field delimiter to delimit data fields in the database. Thus a | character may not be used in the result string. **DFvalidate** will block any attempt to select an entry from a lookup table which contains a | in the result string.

In some lookup tables the search string will be a short acronym and the result string will be a longer description. In other lookup tables (e.g. COSTART) the search string will be a long description, and the result string will be a short term. Either style is acceptable. Also, for some entries the search and result strings may be identical.



Note

A lookup table must be a plain ASCII text file. If it is anything other than this, an error message will appear in a blocking, warning dialog upon starting **DFvalidate**. If **DFbatch** is used, an error message will appear on the command-line upon starting **DFbatch**.

2. **The lookup tables used in each study must be registered in a file, named DFlut_map, and stored in the study lib directory.** Each lookup table is specified on a separate line by providing a lookup table name and the full path name, or relative pathname from the study lib directory, to the file, separated by the | field delimiter (e.g. **DRUGCLASS|/studies/lookups/drugclass.lup**). This makes it easy to maintain standard lookup tables which are used across many studies. It is also possible to attach a lookup table to the query and note fields in QC notes, and to the New Reason Value field in Reason for Data Change records. These lookup tables are defined like any other lookup table, but they must be registered in the lookup table map file using the following specific lookup table names:

QC for the QC note query field (e.g. **QC|QCqueries**)

QCNOTE for the QC note field (e.g. **QCNOTE|QCcomments**)

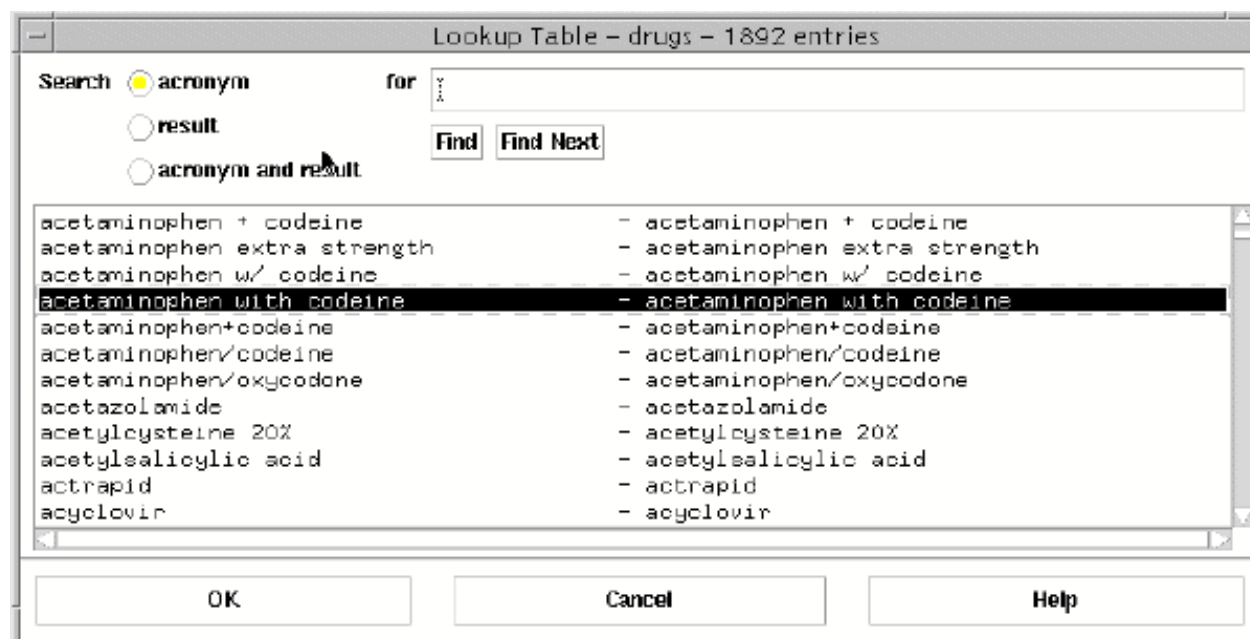
REASON for the reason for data change New Reason Value field (e.g. **REASON|DFreasons**)

3. **To attach a lookup table to a specific data field, the dflookup edit checks function must be attached to the desired field in DFsetup.** . This process is discussed in Edit Checks. This is not required for lookup tables: QC QCNOTE, and REASON, described above. These lookup tables are not applied to specific data fields, but rather to the QC note and Reason dialogs; and can be accessed from an ellipsis button which will automatically appear to the left of the relevant field in the dialog, if the lookup table has been defined.

To use the lookup table, enter a search string in the text field and press Return. If you can't remember the entire search string just enter the first few characters and press Return. If you are really stuck leave the data field blank and press Return. If an exact match is found the result string will be entered into the data field. If more than one possible match is found the lookup table will pop-up on the screen with the first possible match highlighted. The desired entry can then be located and selected using a double click.

The pop-up lookup table is a scrolling window which contains all entries in the lookup table (not just the current possible matches). Both the search string (on the left) and the result string (on the right) are shown for each entry in the lookup table, as illustrated in the following example.

Figure 5.23. The lookup table scrolling window.



If no matches are found, the lookup table will pop-up with the first entry in the lookup table highlighted. To continue your search you can scroll through the list to find the desired entry, and select it using a double click. Alternatively, you can enter another search string in the Search For field at the top of the window, and press Return. Again, if an exact match is found the result string will be entered into the data field, or you can select the desired data field with a double click. If you make a mistake, simply double click again, on the desired entry. Its result string will replace the current entry in the data field.

When you have completed data entry, the lookup table can be closed by selecting the Close button at the bottom of the window or by simply tabbing to the next data field.

5.3.3.6. Flagging the Value of the Current Field as Not Available

If a data field should have something in it but you are informed that this item is unavailable you can mark it NA by choosing a code from the Mark Data NA pullright menu in the Misc menu. Fields marked NA turn the NA field color (green) and the selected NA code is inserted into the data field.

The not available code menu can be configured for a study by creating a small database of missing codes. The database of missing codes must be named `STUDY_DIR/lib/DFmissing_map`, that is, the file must be named `DFmissing_map` and it must exist in the `lib` sub-directory of your study hierarchy. The format of the file is clear ASCII text, one missing code definition per line, each definition having two fields, delimited by a `|`. The first field in each definition is the single character, ASCII code to be inserted into the database and the second field is the textual description of that code. This file must currently be created and maintained with a standard text editor, like `vi`.

Example 5.2. A sample of a `DFmissing_map` file.

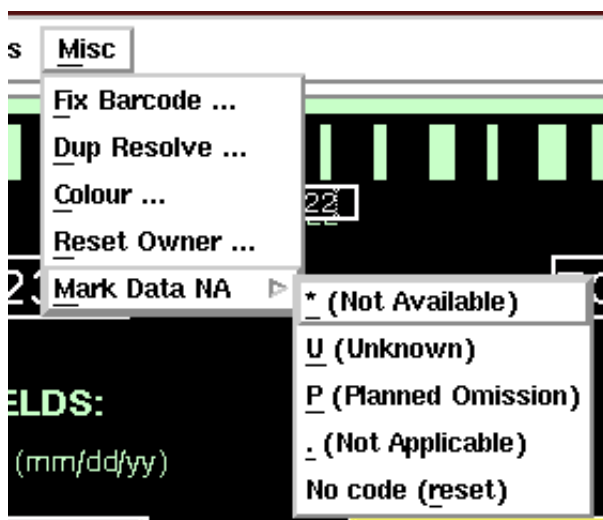
```

*|Not Available
U|Unknown
P|Planned Omission
.|Not Applicable

```

The above example defines four missing codes *, U, P, and ..

Figure 5.24. The corresponding Mark Data NA pullright menu for `DFmissing_map` above.



If the `DFmissing_map` file does not exist, is empty, or cannot be read because of permission problems, then the only code that is available is Not Available with matching database code *.

In **DFvalidate** the codes can be accessed from the Mark Data NA menu which appears under the Misc heading or in the fast menu (recall that you can access the fast menu with the right mouse button anywhere over the top validation tool window). To set the current data field to one of the missing codes, select the appropriate code from the menu. The selected code will be inserted into the data field, the NA color will be set for the field, and the textual description of the code will appear in the middle message window of **DFvalidate**.

As shown in the menu, each code also has a mnemonic that can be invoked by typing **F10-m-m** and the code letter in one sequence. To remove a missing code from the current data field, select the No Code (Reset) menu option.

In the special case where the current data field is a text field, the desired single character code can simply be typed in the field. In this case, however, the field will not be marked with the NA color until the next (or previous) field becomes current.

5.3.3.7. Check Type Data Fields

Check boxes can be set with the mouse or the keyboard. With the mouse, each time a check box data field is selected, the current setting of the check box is toggled, either from off to on, or on to off. With the keyboard, the current setting of the check box can be toggled using the space bar. The value of the check box can also be set by entering the number 1. The value of the check box can be unset by entering the number 0, the BackSpace key, or the Delete key.

The value stored in the database for check fields is zero if blank and one if checked.

5.3.3.8. Choice Type Data Fields

For choice responses, i.e. questions with a few possible answers, use the number keys (1 - 9) to make your choice.

Figure 5.25. A choice question with 3 response options.

☐ *No* ☐ *Yes* ☐ *Uncertain*

The keyboard mapping is by position, with increasing positions moving from left-to-right within top-to-bottom. In this example, position 1 = No, 2 = Yes, and 3 = Uncertain.



Note

The actual value stored in the database for a given choice may be different than its positional value if it has been coded this way during study setup.

The Backspace key, the Delete key and the 0 key each deselect any current choice and set the current database value to 0.

Using the keyboard, a choice can also be made by using the arrow keys to change the highlighted choice field and then pressing the space bar to select the highlighted choice.

Finally, the response value can also be set by selecting the desired choice with the mouse. Any previous choice value is deselected and the new choice is selected.

5.3.3.9. Visual Analog Scale Type Data Fields

The data value corresponding to a Visual Analog Scale (VAS) variable is represented by a numeric data field. The numeric value is the relative position of the scale mark between the endpoints of the scale. In most cases the ICR software is able to correctly read all VAS values and insert the correct numeric value into the data field.

You can modify the value in the current VAS field in 2 ways:

1. Simply type in a new numeric value (unless the scale is very coarse this is very hard to do and error prone).
2. Use the mouse to specify a new value. Position the mouse in the CRF window. You will notice that the message window now displays the message *To read VAS: select left end, slash mark, and right end*. Position the mouse exactly over the left endpoint of the VAS and click with the left mousebutton. Now position the mouse exactly over the slash mark, and again click left. Finally, position the mouse over the right endpoint and click again. The message window will display a message similar to *VAS reading is 45 (was blank)* and the new value will appear in the data field of the data window. This is confirmation that your mouse selections have been interpreted correctly. If you make an error at any point, simply move the mouse cursor outside the CRF window, and enter the window again to restart the process. Currently, only horizontal visual analog scales are supported and hence only the horizontal position of the mouse cursor is relevant any time a point is being marked.

5.3.3.10. Date Type Data Fields

Date values can be entered simply by typing the desired numbers or characters into the current date field. Component delimiters are automatically inserted into the correct positions as defined by the date format in **DFsetup**. Because date fields must accommodate 3-character month values, DataFax allows any legal alphanumeric character to be entered into each of the day, month, year components. Upon exit of the date field, sensibility testing is performed by the system to identify illegal or non-sensical date values. These problem values will cause the field to turn red making them easy for the user to identify.

5.3.4. Signing Off Records: Setting Status and Validation Level

When a CRF page has been reviewed, data entry has been completed, changed data values requiring reasons have been documented, and any problems have been flagged with QC notes, the record must be signed off by selecting a status in the last data field at the bottom of the data entry window. The available statuses are:

- **clean**: no problem data
- **dirty**: there are one or more problems on the page which need to be corrected
- **error**: it was not possible to determine one or more of the keys (patient ID, plate number or visit/sequence number) for the page.

When this is done the status and validation level of the record are updated to the values that the user is attempting to assign and all changes made to the data record are committed to the database. **DFvalidate** then scrolls to the next record in the retrieval set. Signing off a data record must be done to save changes at all validation levels.

When working in **Edit** or **Validate** modes you must sign off each record by setting its status to keep any changes you have made to the record (including the addition or modification of QC notes, and changes to reason records). Your changes are recorded in the database immediately. Moving to another record without setting its status, quietly ignores all changes and returns that record to its former condition. This can be used as a method of backing out of changes which you decide you do not want to keep.

You must also sign off records to promote them to your working validation level when working in **Validate** mode. Even if you have not changed anything on the record, remember to set its status or it will remain at its current validation level.

Record status can be set through a variety of methods using either the mouse or the keyboard. When the record status is set for a new record, both the creation and modification data fields are automatically filled in by **DFvalidate**. When the record status is set for a previously validated record, the modification timestamp is updated if there were any changes to the data.

The methods for setting a record status are as follows (you can use any method you prefer):

- When you reach the bottom of the page and the current data field is the Status field, typing **1**, **2** or **3** sets the status to **clean**, **dirty** or **error** respectively.
- Make the status visible in the scrolled data window and use the mouse to select the desired status.
- The mnemonics can be used at any time during record validation, i.e. the status field does not have to be the current variable, in fact, it does not even have to be visible. As a reminder, all menu mnemonics are accessed starting with the F10 key. Then the underlined letter in the desired menu is typed (r for record in this case), and finally the underlined letter for the desired menu item is typed (c for clean, d for dirty, or e for error).
- You can also set status at any time by choosing the desired status from either the Record menu, or from the fast menu available by holding down the right mouse button anywhere in the data window.

You cannot set record status to `clean` if the record has illegal values, unresolved QC notes, or missing key fields. Attempts to do so will generate a complaint printed in the message window, and the current field will move to the first field on the page with an illegal value or unresolved QC note. Records with missing keys (visit or patient ID) must be set to status `error`. Records with illegal values and/or unresolved QC notes must be set to status `dirty` or `error`.

There are however 2 conditions under which a record containing an illegal value can have its status set to `clean`.

1. The most common, occurs when a QC note has been attached to a data field to flag a problem, and then is subsequently resolved (eg. because the investigator confirmed that the value, although unusually high or low, was indeed correct).
2. This condition occurs when working at the highest validation level (level 7). At this level, records containing illegal values can be forced to status `clean` by the user (after confirming this action in a dialog) without requiring that each illegal value have a corresponding resolved QC note.



Note

An exception to this condition occurs when one or more of the key fields are illegal because they are blank. Regardless of validation level, no user is able to assign a record with missing keys a status of anything other than `error`.

Although a user may sign off the record with a particular status, the status of the record may still change after plate exit edit checks have executed. This will occur if a user is attempting to assign a `clean` status to a record which is `dirty`. If necessary, the original plate enter values for record status and validation level may be retrieved using the `dfvarinfo` edit check function described in *DataFax Programmer Guide*, `dfvarinfo`.

5.3.4.1. Record Sign Off and QC Notes

New QC notes and modifications to existing QC notes are saved upon selecting the OK button in the QC dialog, but they are not committed to the database until the data record to which they are attached is signed-off. If the data record is not signed-off, then the changes are lost.

QC note deletion is treated differently. When a QC note is deleted a dialog appears requesting confirmation. If given, the QC note is immediately deleted from the database and the field color (previously associated with the QC note) is replaced by the color associated with the current data value (i.e., legal, illegal, optional, or missing). The data record need not be signed-off in order to delete QC notes.

5.3.4.2. Record Sign Off and Field Value Shortening

Rarely, a variable definition may be modified in **DFsetup** such that its length is decreased or its format altered. When this occurs, **DFvalidate** checks each field's value against its variable definition and shortens any value that exceeds the defined maximum (See Validation of Old Records Following a Setup Change). Signing off a record that contains a shortened field value will save and store the shortened value in the database. If the record is not signed off, no changes to any of the data fields will be saved, and thus any pre-existing mismatch between a data value and its field format will remain in the database.

5.3.5. Resolving Duplicates During Validation of New Records

Pages of the CRF may be faxed more than once, either unintentionally or because of a refax request made on a QC note attached to the data record. Duplicates are resolved by marking one version as the primary record. Other versions can be retained as secondary records. There can only be one primary record, but there can be as many secondary records as required. The primary record will typically be the

last copy faxed and will contain some additions or corrections that were not present on any previous copies received.

The primary record is the only data record which can have attached QC notes. Whenever a new faxed CRF page is set to be the primary version, the existing primary version is changed to secondary and any QC notes which may have been attached to it automatically migrate to the new primary record.

By retaining all versions of each CRF page you can keep a record of all corrections and additions made at the clinical sites. You can think of this as equivalent to having all versions of a CRF page clipped together, with the most up to date copy on top, and any QC notes moved to the top copy, to indicate both problems which are outstanding and problems which have been resolved.

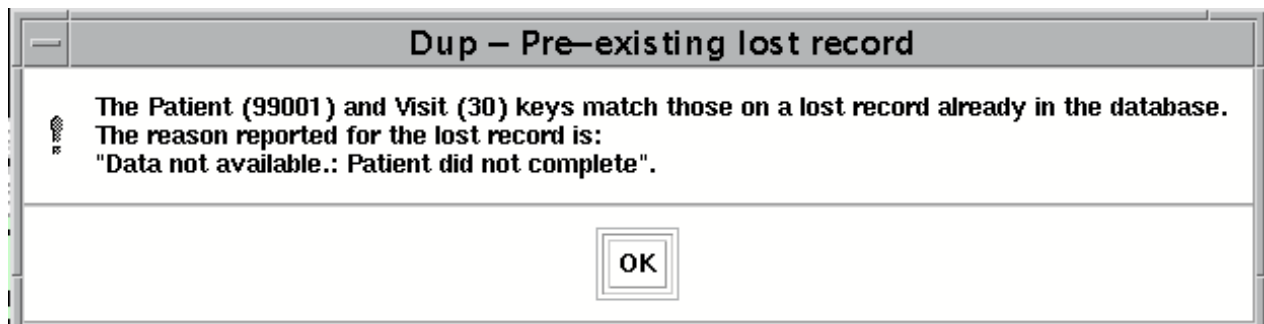
Duplicates are detected in two ways.

1. A push button exists in message bar between the data and CRF window of the split screen. A number inside the push button tells the user the number of records that exist for the current record. If the number is 1, only a primary record exists. By clicking on the push button, the user can open the Dup Resolve dialog and review the duplicate and primary records.
2. Duplicates are also detected when new records are being validated, as soon as the patient ID number has been validated.

5.3.5.1. Duplicates and Lost Data Records

If the current data record matches, on all 3 keys, an existing record in the Lost Data Log, the following dialog box will appear.

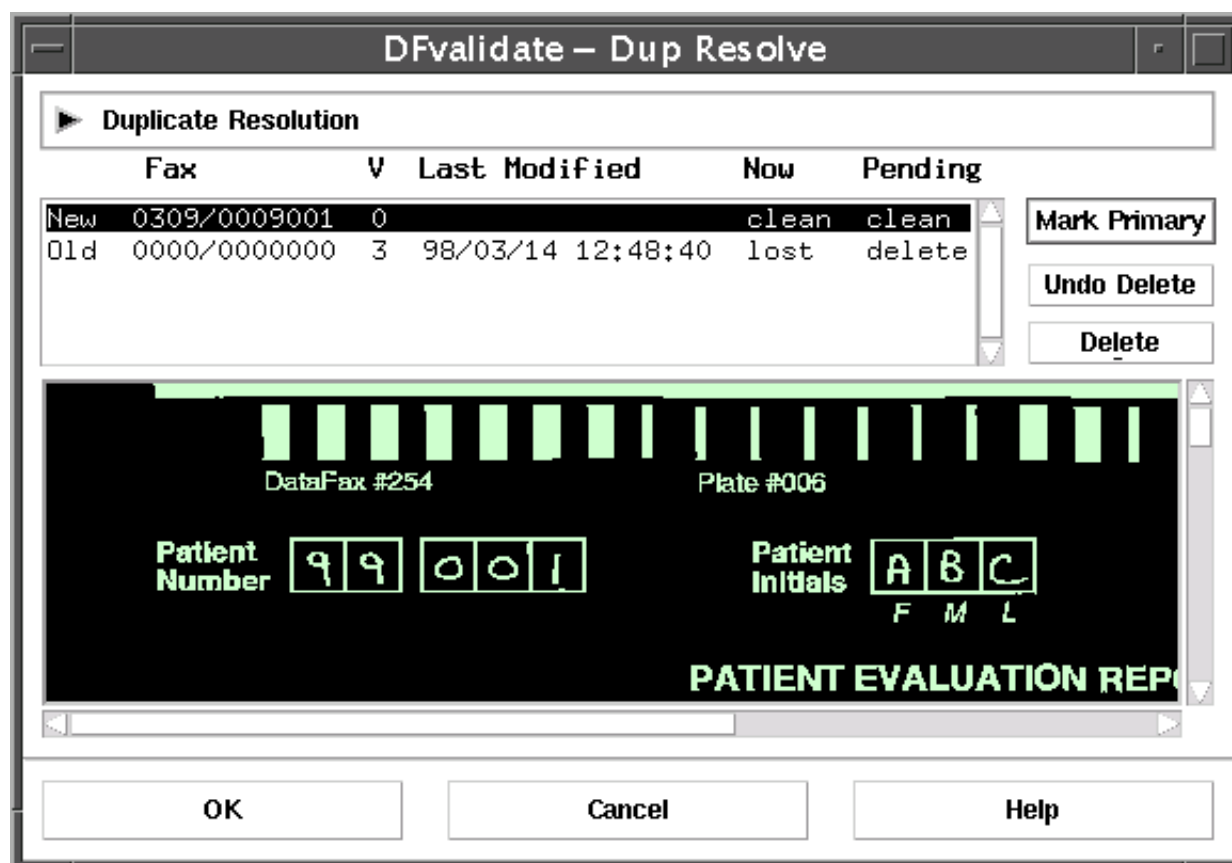
Figure 5.26. The Dup - Pre-existing lost record dialog



The pop-up dialog warns the user that the record being entered matches a lost record already in the database. There are no decisions to be made at this point because the duplicates between the lost record and the current record are handled at record sign-off in the normal duplication resolution manner. However, the warning is present so that users who do not wish to deal with the duplicate resolution (or are not permitted to) can abort processing of the current record and immediately proceed to the next one.

At record sign-off, the following duplicate resolution dialog box will appear.

Figure 5.27. The Dup Resolve dialog for a lost record

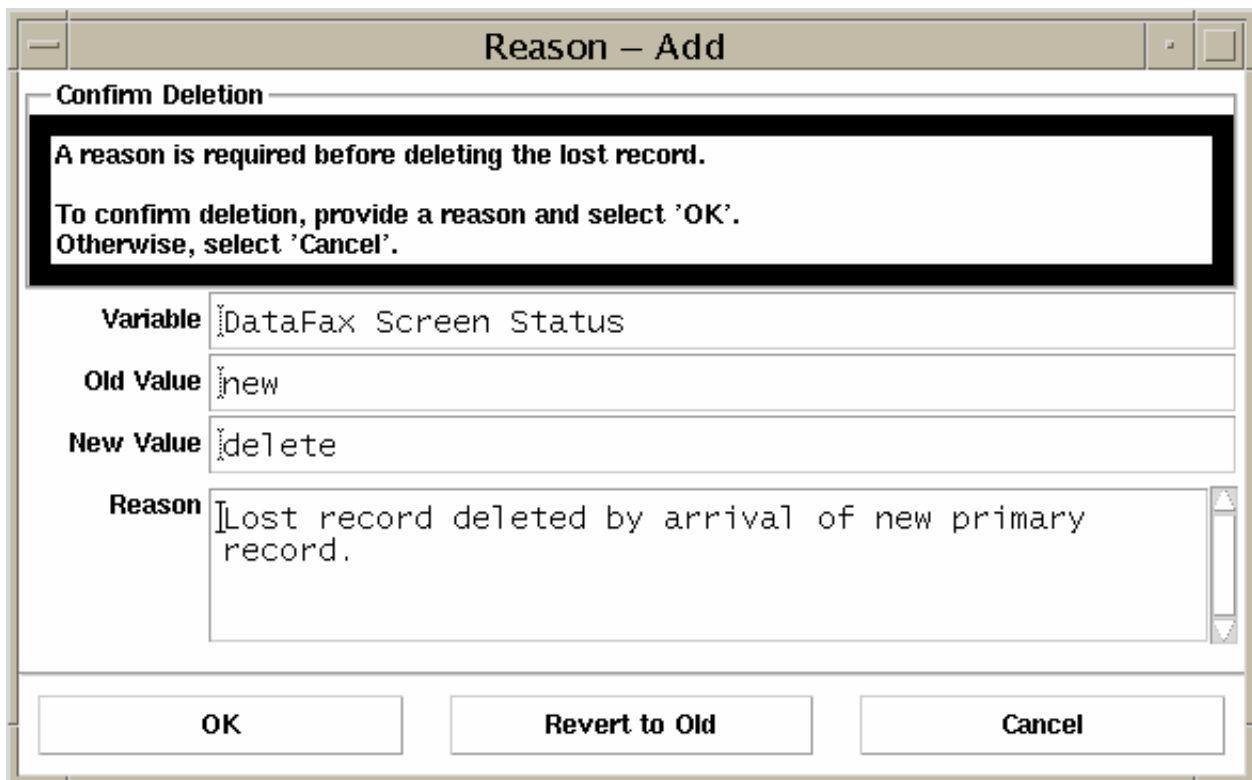


At this point, there are 2 options:

- select OK, the default option, to delete the lost record and keep the current record as the new primary record.
- select Cancel to abandon duplicate resolution and return to the current data record. In this case, the current data record can be skipped so that duplicate resolution may be handled at another time, or by another user.

If a user chooses to delete a lost record during the duplicate resolution process, a reason for data change is required for the deletion. Once the OK is selected, the Reason - Add dialog appears as follows.

Figure 5.28. The Reason dialog for the deletion of a lost record.



Reason - Add

Confirm Deletion

A reason is required before deleting the lost record.

To confirm deletion, provide a reason and select 'OK'.
Otherwise, select 'Cancel'.

Variable: DataFax Screen Status

Old Value: new

New Value: delete

Reason: Lost record deleted by arrival of new primary record.

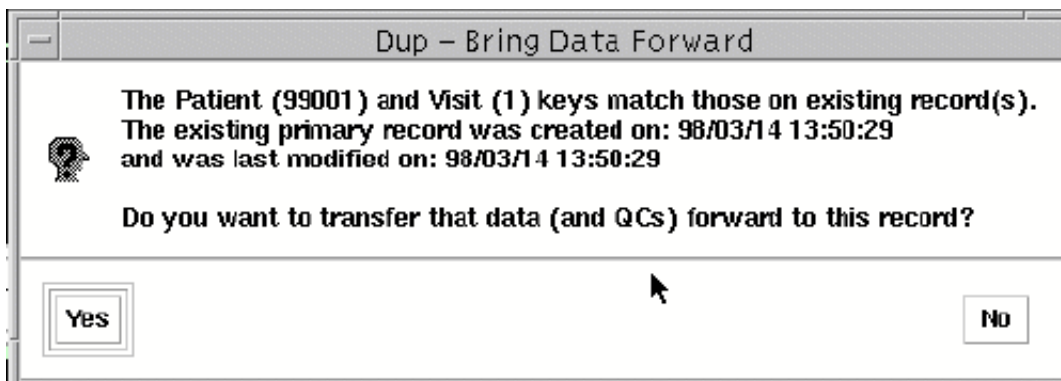
OK Revert to Old Cancel

In the dialog, the New Reason Value field is pre-filled with the default reason text `Lost record deleted by arrival of new primary record..` The user has the option of either editing or keeping the default reason text, and selecting OK in the reason dialog to commit the reason for data change and complete the duplication resolution process. If the user cancels the Reason - Add dialog without supplying a reason, the commit action for the duplicate resolution process is also cancelled.

5.3.5.2. Duplicates and Existing Data Records

If the current data record matches an existing one on all 3 keys (patient ID, visit and plate numbers) the following dialog box will appear.

Figure 5.29. The Dup - Bring Data Forward dialog



Dup - Bring Data Forward

The Patient (99001) and Visit (1) keys match those on existing record(s).
The existing primary record was created on: 98/03/14 13:50:29
and was last modified on: 98/03/14 13:50:29

Do you want to transfer that data (and QCs) forward to this record?

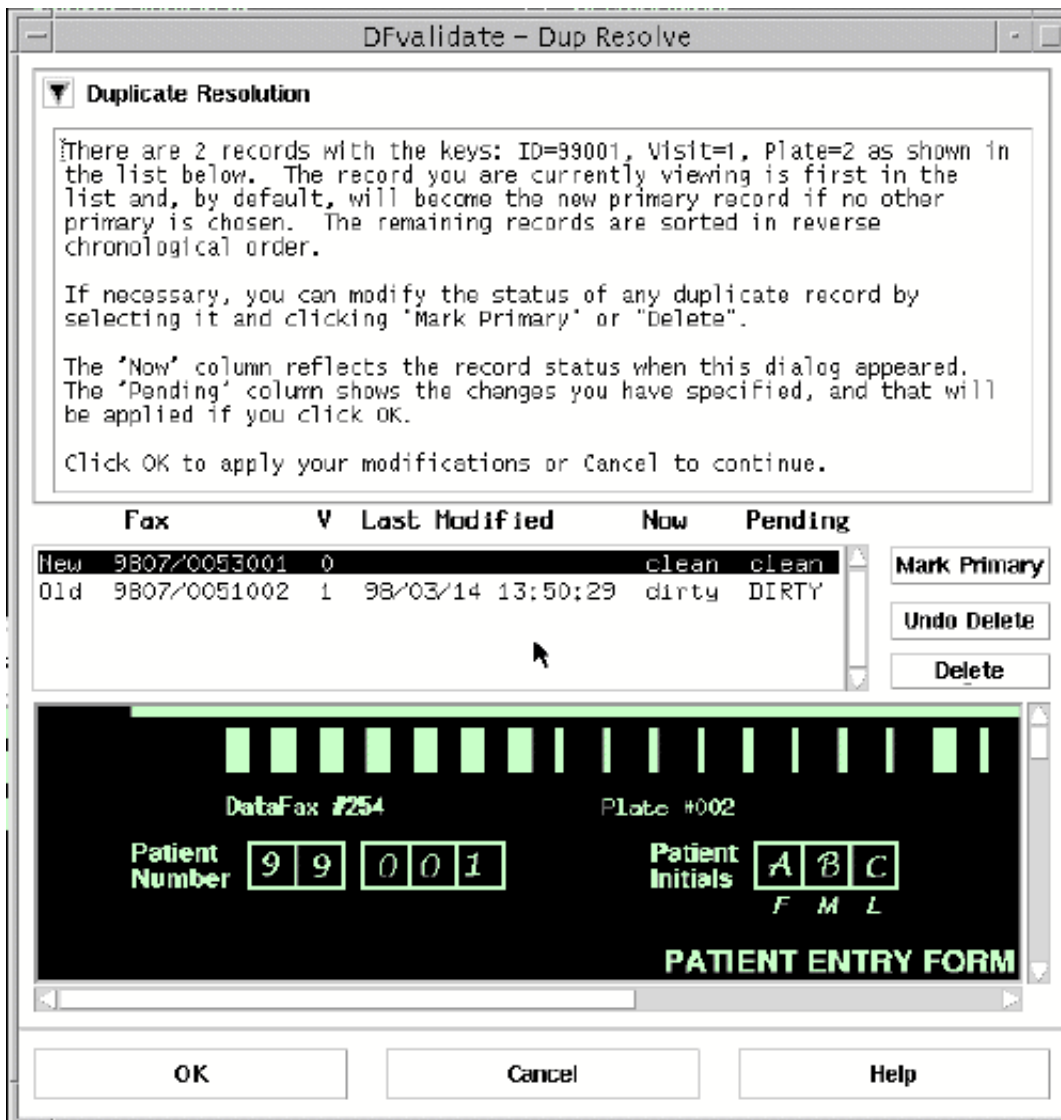
Yes No

The pop-up window identifies when the existing primary record was created and last modified and asks if you wish to bring the data and QC notes forward from that existing primary record into the current data entry window. Typically you will want to answer **Yes**, and then review all of the data fields against the new CRF image. Hopefully you will find some new information and QC notes which can now be resolved.

The QC notes which are brought forward must be resolved manually, with the exception of QC notes of type missing or illegal. Missing and illegal QC notes may be resolved automatically if the Preferences settings "Resolve missing value QCs as data is corrected" and "Resolve illegal value QCs as data is corrected" have been selected in the Preferences dialog in **DFvalidate** (see Preferences). It would be unwise for DataFax to automatically resolve old QC notes of any other type when a corrected page arrives because it is possible that the original problem was not completely resolved. In such cases you will want to reissue the old QC note(s), perhaps with a revised explanatory query. It is very important to remember to examine, and if possible to resolve, the QC notes. If this is not done, the data record can not be signed off as clean and any unresolved QC notes will reappear on subsequent QC reports.

When you reach the bottom of the page, and sign off the record by setting its status to `clean` or `dirty`, the following duplicate resolution dialog box will appear.

Figure 5.30. The Dup Resolve dialog



At this point you can accept the new record as the primary version of this CRF page by selecting OK. This is the default resolution. However, if you are unsure of the resolution (e.g. because no QC notes were brought forward and the new CRF image did not appear to contain any new information), you can review the individual fax images. You can do this by selecting the image of interest from the scrolling list of images.

Figure 5.31. The image of the selected entry in the Dup Resolve window is displayed in the scrolling window.

	Fox	V	Last Modified	Now	Pending
New	9807/0053001	0		clean	clean
Old	9807/0051002	1	98/03/14 13:50:29	dirty	DIRTY

Buttons: Mark Primary, Undo Delete, Delete

Patient Number: 99001

Patient Initials: ABC

PATIENT ENTRY FORM

Randomization

Provided the patient meets all eligibility criteria, call the randomization hot line for random medication code number.

Medication Code #: 1143

Buttons: OK, Cancel, Help

You can collapse the Duplicate Resolution message to increase the space available to the image window by clicking the arrow button, or you can resize the Dup Resolve dialog bigger so that more space is available for the image window.

To mark any record as the primary at this point, select the appropriate entry from the scrolling list. Make sure that the entry is highlighted in the scrolling list, then click Mark Primary. The Pending status of the new version is set to primary and the Pending status of the previous primary is changed to secondary. Whether the new record is marked `clean`, `dirty` or `error` is determined by how you signed off the page just before the Duplicate Resolution dialog box appeared and whether or not any existing QCs will be migrated from the previous primary record.

If you subsequently realize that you made an incorrect choice of primary record, you can still highlight another entry from the scrolling list, and click Mark Primary. This will again change the Pending status of the existing primary to secondary and the highlighted record to primary.

The Pending status column reflects all of the changes that you have specified. These Pending changes are applied to the database only when the OK button is clicked.

Any or all secondary versions can then be deleted by highlighting them and then clicking Delete. DataFax will not allow you to delete the primary record within the Dup Resolve window.

Duplicates can be resolved by all users, regardless of their working validation level. Thus a data clerk working at validation level 1 can mark a newly arrived page as the primary version even if the existing primary record is at a higher validation level. This is necessary to allow a CRA to put a refax request on a page and then to allow the data clerk to accept the refaxed CRF when it arrives. In this case the new primary record will be set to validation level 1 and will again go through all of the subsequent validation steps for the study, thereby maintaining the integrity of your workflow.

We recommend that a data manager or CRA review all data records at data entry validation levels (typically 1-3) which have attached QC notes, both to review unresolved QC notes before they are issued on future QC reports and to review any QC notes resolved through the receipt of a refaxed CRF page.

5.3.6. Resolving Duplicates During Validation of Old Records

Duplicates can be examined (and a different version selected as primary) at any time during record validation by choosing Dup Resolve from the Misc menu. This will bring up the Dup Resolve dialog previously described. Thus if a data manager or CRA saw that a data entry clerk had accepted a new fax as the primary version and resolved a QC note, the old version could be examined by selecting Dup Resolve. Also, if you have marked a new record as primary and kept the old version as secondary and then discover that you have made an error you can retrieve these records and change the secondary record back to primary.

Old secondary records can be deleted at any time by retrieving them and choosing Delete from the Record menu. If you make an error just reset the record status to `CLEAN`, `DIRTY` or `ERROR` before you release the set. A record marked for deletion is not actually deleted from the database until the record set is released, giving you an opportunity to correct any mistakes in specification.



Note

It is not possible to delete the primary record when there are one or more secondary records. In such a situation, the Dup Resolve dialog must be used to choose a new primary record first and then the desired record can be deleted.

5.3.7. Validating Returned QC Reports

The creation and use of QC (quality control) reports is described in DataFax Quality Control Reports. We will only mention here that when the clinical sites fax back answers to questions by faxing back the report itself, you will see the returned report in **DFvalidate**. All that is required at this stage is to type the report number in the data entry window (without the dash which separates the center number and creation date) and sign the page off as clean. This will store the returned QC report in the study database (just like any other plate) from which it can later be retrieved and reviewed or printed using the CRF Viewer Tool (**DFviewer**) or **DFvalidate**. A suggested standard operating procedure for making corrections to the database from these returned QC reports is described in Resolving QC Notes on Returned Quality Control Reports.

5.3.8. Validation Using Double Data Entry

DataFax Double Data Entry (DDE) can be performed in 2 ways:

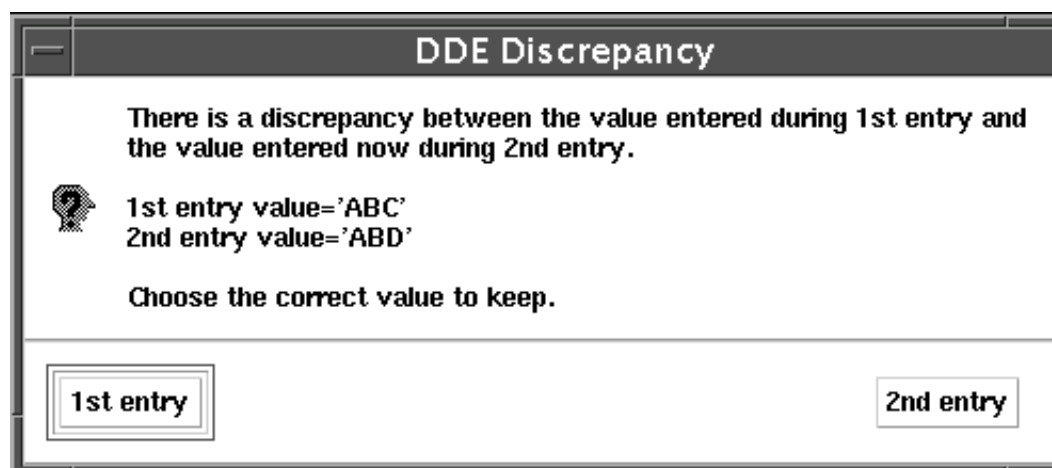
- heads-up double data entry
- double data entry requiring third-party arbitration.

The two methods are not mutually exclusive. Heads-up DDE is performed in real time so that the second reviewer immediately sees, and has the ability to resolve any differences in the entry. Following heads-up DDE, DDE programs, executed from a command line, perform a comparison of the data and generate printable discrepancy reports. Generating comparisons and reports constitutes the optional, third-party arbitration method. What follows is a discussion of heads-up DDE. The method for creating DDE record sets and third party arbitration functionality are described in *DataFax Programmer Guide*, *DFdde: DataFax Double Entry*.

Heads-up DDE requires the creation of record sets on which DDE is to be performed. Creation of these sets are performed with the **DFdde.set** command, as described in *DataFax Programmer Guide*, *DFdde: DataFax Double Entry*. To initiate DDE, the DDE record set is chosen using the By DDE Batch File retrieval method described earlier in this chapter. During DDE, the user tabs through the data fields on the current page and enters the values as they would during normal entry. As each field is exited, the normal end of field processing occurs (i.e., reason for data change and edit check functionality). If the first and second entry values do not differ upon exit of the field, nothing else is required and the user is able to continue on with second entry and eventually record sign-off.

If the first and second entry values are discrepant, the user is presented with a question dialog identifying the first and second entry values.

Figure 5.32. DDE discrepancy dialog box.



If the user chooses to keep the first entry value, nothing else is required. If the user chooses to keep the second value, the reason for change dialog immediately appears and a reason for change is required before continuing to the next field on the page.

Once DDE has been performed on a data record, the primary record is updated by the DataFax server and saved to the database. Hence, if the record has been modified in any way, the primary record will be written to the relevant study journal files and the changes can be traced using the **DF_ATmods** program in the DataFax **DFreports** tool. Records that have undergone a second review will have their validation levels updated to match that of the second reviewers validation level.

Following the completion of heads-up DDE, the user has the option of generating a record comparison and creating discrepancy reports that may be reviewed by a third party individual.

5.4. Validation Tool Menus

The preceding sections of this chapter have described the main operational steps in using **DFvalidate**. The remainder of this chapter describes each of the functions from the **DFvalidate** menus.

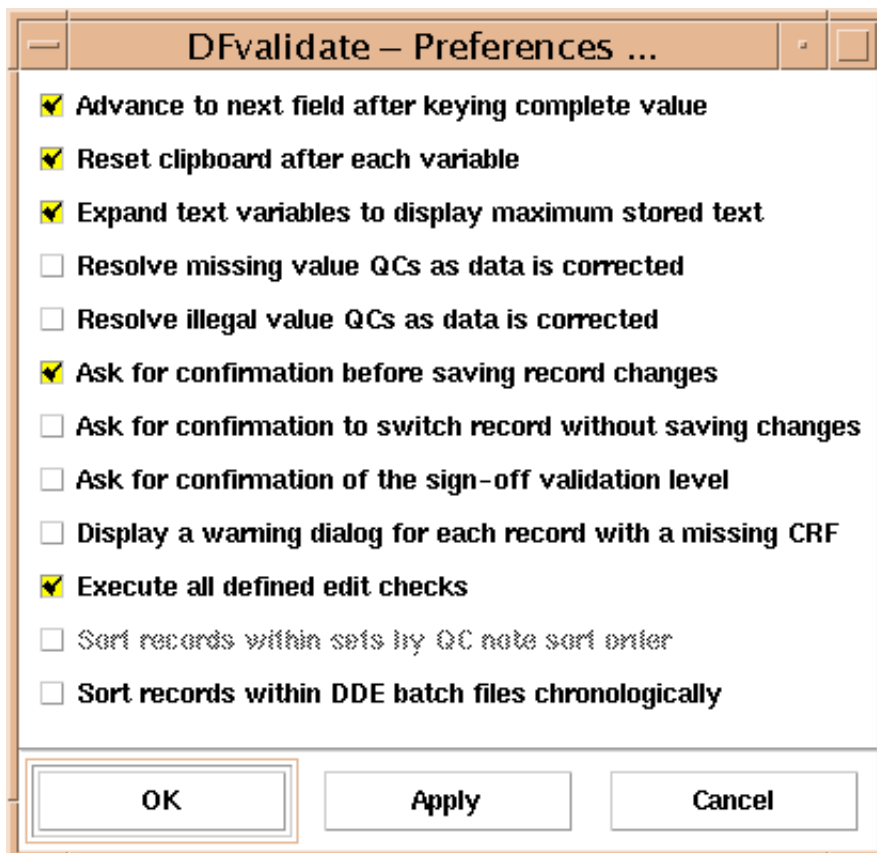
5.4.1. File Menu

The File menu contains the > Preferences menu item and the Exit menu item.

5.4.1.1. Preferences

There are various points in **DFvalidate** where two possible types of behavior would effectively achieve the same result. Since each user may have different preferences about which type of behavior is better suited to their own work procedures, the Preferences dialog allows them to tailor the behavior of **DFvalidate**. When **DFvalidate** starts, the preference settings are obtained from the `app-defaults` specifications. Subsequently, these preference settings can be modified for the duration of time that the user has **DFvalidate** in use. This is done with the Preferences dialog. Permanent changes must still be made through the `app-defaults` mechanism.

Figure 5.33. The Validation Tool Preferences dialog



Each of the preference settings is simply a toggle, to turn on or turn off a particular type of behavior. To modify the preferences for the current session of **DFvalidate**, make your desired changes to the toggle buttons. When you are satisfied with your settings, click OK to apply your changes and pop-down the Preferences window. Alternatively, click Apply to apply your changes but leave the Preferences window visible. To discard any changes and pop-down the window, click Cancel.

The meaning and behavior of each preference is described below.

- **Advance to next field after keying complete value.** Some users may prefer to not have to use the Tab or Enter keys to move between all data fields. If this preference is turned on, **DFvalidate** will automatically move to the next field when the keyboard is used to complete the value of the current field.
- **Reset clipboard after each variable.** When the clipboard is visible, it contains any text that has been copied into it from a data field. When the current field advances to the next data field, the default behavior is for the value previously displayed in the clipboard to remain visible. If you enable this preference, you instead clear out the contents of the clipboard each time the current field advances.
- **Expand text variables to display maximum stored text.** This preference controls whether or not text fields grow to the size of the maximum stored text when they become current. The text field (and its value) appears as a multi-line text widget in place of the current data text widget if the number of characters storable in the widget is greater than the number of characters displayable.
- **Resolve missing value QCs as data is corrected.** This preference enables/disables the validation tool's ability to automatically resolve QC notes for missing values when the data is corrected.

When enabled, the following occurs at each field:

1. At field entry, the current value in the data field is evaluated for missing status.
2. At field exit, and before any field exit edit checks are executed, ^[7] the value is again evaluated for missing status.
3. The QC note is automatically resolved if all of the following conditions are met:
 - there is an unresolved missing value QC note on the field,
 - the field was empty upon entry to the field,
 - the field is not empty upon exit from the field (this includes the presence of a missing value code), and
 - the auto resolve missing value QC note preference is enabled.

When a missing value QC note is auto-resolved, the status of the QC note is set to Resolved, Corrected, and the current user's name and date/time stamp are entered in the resolution timestamp field. If the data field now contains a missing value code, the resolution status is set to Resolved, NA.

By default, this preference is disabled so that missing value QC notes will not be resolved automatically.

- **Resolve illegal value QCs as data is corrected.** This preference enables/disables the validation tool's ability to automatically resolve QC notes for illegal values when the data is corrected.

When enabled, the following occurs at each field:

1. At field entry, the current value in the data field is evaluated for illegal status.
2. At field exit, and before any field exit edit checks are executed, ^[7] the value is again evaluated for illegal status.
3. The QC note is automatically resolved if all of the following conditions are met:
 - there is an unresolved illegal value QC note on the current field,
 - the field was illegal upon entry to the field,
 - the field is not illegal upon exit from the field, and
 - the auto resolve illegal value QC note preference is enabled.

**Note**

Because illegal value QCs are only resolved when an illegal value is changed to a legal value it is important to note that a blank is considered legal if the field is optional and illegal if the field is required or essential, and a missing value code is legal for optional or required fields.

When an illegal value QC note is auto-resolved, the status of the QC note is set to Resolved, Corrected, and the current user's name and date/time stamp are entered in the resolution timestamp field. If the data field now contains a missing value code, the resolution status is set to Resolved, NA.

By default, this preference is disabled so that illegal value QC notes will not be resolved automatically.

- **Ask for confirmation before saving record changes.** When you make changes to the data in a previously validated record, **DFvalidate** asks you to confirm your modifications before sending them to the database server. If you would rather that it not ask for confirmation each time, disable this preference.
- **Ask for confirmation to switch record before saving changes.** By default, **DFvalidate** allows you to move between records without requiring that you save your changes to each record. When you leave a data record without saving your changes the changes are lost. This applies to changes to data fields, QC notes and reasons for data changes. To commit your changes to the database you must remember to set the record status to `clean`, `dirty` or `error` before you leave the data record. If you fail to do this you will lose all changes when you: select another record from the record list, select another record using the page turning buttons, or with the Record Previous and Record Next menu items), release the current set of records, switch to another set of records, merge existing sets to form a new set of records, or proceed to retrieve a new set of records from the database. To prevent this from happening, enable this preference, and you will be reminded each time you attempt an action that would leave a data record which has data, QC or reason changes that you have not saved.
- **Ask for confirmation of the sign-off validation level.** If this preference is clicked on, the user is asked to confirm (or modify) the sign-off validation level of the current record whenever it is signed off to the database. This can be useful in the case where most records are to be validated to level 'X' but the occasional record is to be validated to level 'Y'. When this preference is on, the following dialog appears as soon as each record is signed off:

Figure 5.34. The Setting Record Status dialog appears upon record sign-off if the Ask for confirmation of the sign-off validation level preference is turned on.



- **Display a warning dialog for each record with a missing CRF.** With this preference enabled, **DFvalidate** will display a warning dialog similar to the one below whenever it is unable to locate the CRF image for the current record. This should be a rare occurrence.

Figure 5.35. The CRF Image Load dialog box.



- **Execute all defined edit checks.** With this preference enabled, all edit checks that have been defined for the study will be executed when they are triggered during data field traversal. It is recommended that edit checking be enabled during **DFvalidate** usage, however, if situations arise where an edit check performs unexpectedly, you may want to temporarily disable this preference.
- **Sort records within sets by QC note sort order.** This preference enables/disables sorting of retrieved records in the order specified by the study QC note sort order file `DFqcsort`. If no `DFqcsort` file is defined, the existing sort order of ID, then visit, then plate (default sort order) is maintained and the preference cannot be enabled. However, if the `DFqcsort` file is present, and the preference is enabled, records are sorted for presentation in the sort order defined by the file. Enabling the preference affects the subsequent sets that are built but has no effect on the current, if any, set. Similarly, disabling the preference affects the subsequent sets that are built but has no effect on the current, if any, set. At the user level, the `.xdefaults` file can be modified by the DataFax System Administrator to permanently toggle on this option as the user's default.
- **Sort records within DDE batch files chronologically.** This preference allows Double Data Entry (DDE) record sets to be sorted chronologically in the record list in **DFvalidate**, overriding the DataFax standard ascending order of patient id, visit, then plate.

5.4.1.2. Exit

The Exit menu item is used to quit **DFvalidate**. If you still have record sets in use when you select the Exit menu item, you will see the following warning dialog.

Figure 5.36. A warning message appears if you attempt to exit DFvalidate without first releasing your record set(s).



Choosing Release & Quit releases each of the record sets that is still active and then exits **DFvalidate**. Conversely, choosing Cancel returns you to **DFvalidate**, cancelling your File > Exit action.

5.4.2. Set Menu

The Set menu contains all of the functions used to manipulate sets of data records.

5.4.2.1. Build

The Build function, used to build sets of data records, has been described in detail in Retrieving Data Records. Each time this action is executed and applied a new retrieval set is created. Retrieval sets are given unique names by **DFvalidate** so you can easily switch between multiple sets.

5.4.2.2. Release

This function releases (and possibly unlocks) all records in the current retrieval set and removes any memory structures allocated to the current retrieval set. A new retrieval set is chosen from the list of remaining sets, if any.

5.4.2.3. Goto

Choosing the Goto menu item displays a window with a scrolling list of the active retrieval sets and the current set highlighted. You can make another set the current set by double-clicking on its entry in the scrolling list. Use the Close button to popdown the Goto window.

Figure 5.37. The Goto dialog



Warning

All unsaved changes will be lost when switching sets using the Goto feature. Thus, remember to sign off the modified data record(s) before switching sets.

5.4.2.4. Print

Occasionally, you may find it desirable or necessary to document your current record set. For example, a database manager might use this feature to keep documentation on the data validations performed by each data clerk. Print can be used to print a summary of the current record set in a format similar to that displayed in the Record list window.

To print your current record set:

1. Choose Print from the Set menu.

This will open the standard DataFax Print dialog.

2. Select your print options from the Print dialog.

You may want to change Print Filter, Copies, or Options, but you cannot change Page Range.

3. Click OK to print.

Figure 5.38. Output from Set > Print appears as follows.

```
#Created by DataFax for "DataFax Acceptance Test Kit"
#Set created on Wed Jan  3 22:20:52 2001 by eric
#Retrieval mode: Validate
#Level: 1
#
#
```

#	Id	Visit	Plate	Image	Status	Level
	99001	0	1	9807/0051001	clean	1
	99001	1	2	9807/0051002	dirty	1
	99001	1	3	9807/0051003	clean	1
	99001	1	4	9807/0051004	clean	1
	99004	0	1	9807/0052001	clean	1
	99004	1	2	9807/0052002	dirty	1
	99004	1	3	9807/0052003	clean	1

5.4.2.5. Validate

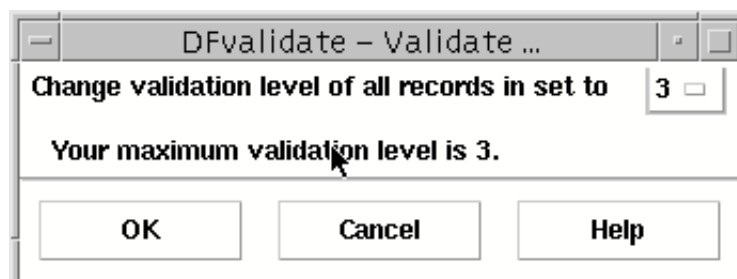
Occasionally you might want to raise or lower the validation level of all records in the current retrieval set. This is accomplished by choosing Validate from the Set menu. You can only raise the validation level to whatever your maximum level has been set at for the current study that you are working on. This may be higher than the validation level you specified in the retrieval criteria for the set. You can lower the validation level to a minimum level of 1.

To change the validation level for the current set of records:

1. Choose Validate from the Set menu.

The following confirmation dialog is displayed. The validation level to change to will be initialized to the validation level that you specified in the retrieval mode for the set.

Figure 5.39. The Set > Validate confirmation dialog.



2. If necessary, change the value for the new validation level.

Do this by clicking on the Change validation level of all records in set to button.

3. Click OK.

If you chose to stop the set validate process, select Cancel.

5.4.2.6. Merge

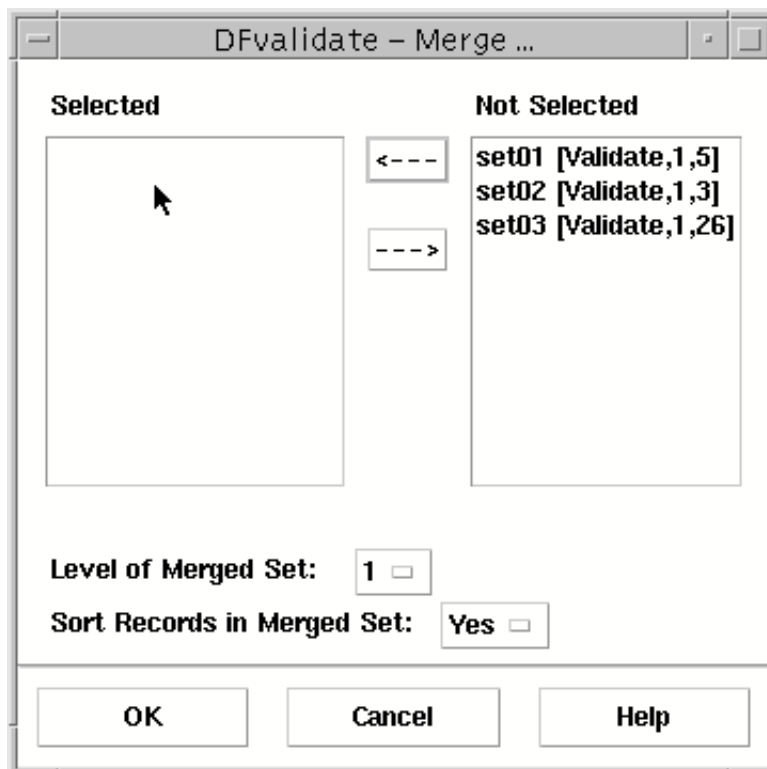
It may be desirable when you have constructed two or more record sets, to merge the two sets into one so that you do not need to repeatedly choose Goto from the Set menu. When the record sets are merged, all records from the specified sets appear in a new set and the specified sets are removed. During the merge process, you can specify whether the records should simply be concatenated together without sorting or whether they should be merge sorted.

To merge two or more record sets into one new set:

1. Choose Merge from the Set menu.

The Merge dialog is displayed. Each active record set will appear in the Not Selected list. Each record set is described by its name, its retrieval mode, and its validation level. When merging sets, **DFvalidate** requires that each chosen set has the same retrieval mode.

Figure 5.40. The Merge dialog. Active record sets appear in the Not Selected list.



2. Select the record sets that are to be merged.

To merge a record set, simply double-click on its name in the Not Selected list, or alternatively, single click on the name in the Not Selected list and click '<---'. The set will appear in the Selected list.

To remove a set from the Selected list, double-click on its name in the Selected list, or alternatively, single click on the name in Selected list and click '--->'. The set will appear in the Not Selected list.

At least two sets must appear in the Selected list and they must have the same retrieval mode.

3. Select Level of Merged Set for the new set.

You can select a validation level that is higher, lower, or the same set as the sets to be merged, but it is an error to select a validation level that is higher than your permitted maximum.

4. Indicate whether or not the records should be sorted when added to the new set.

If sorting is chosen, the records will be sorted in the DataFax sort order by ID, visit, and then plate. If sorting is not chosen, the existing record sets will simply be concatenated together to form the new set.

5. Click OK.

The merge operation will execute and the merged set will become the new current record set. The merged sets are removed from the list of sets.

5.4.2.7. Save

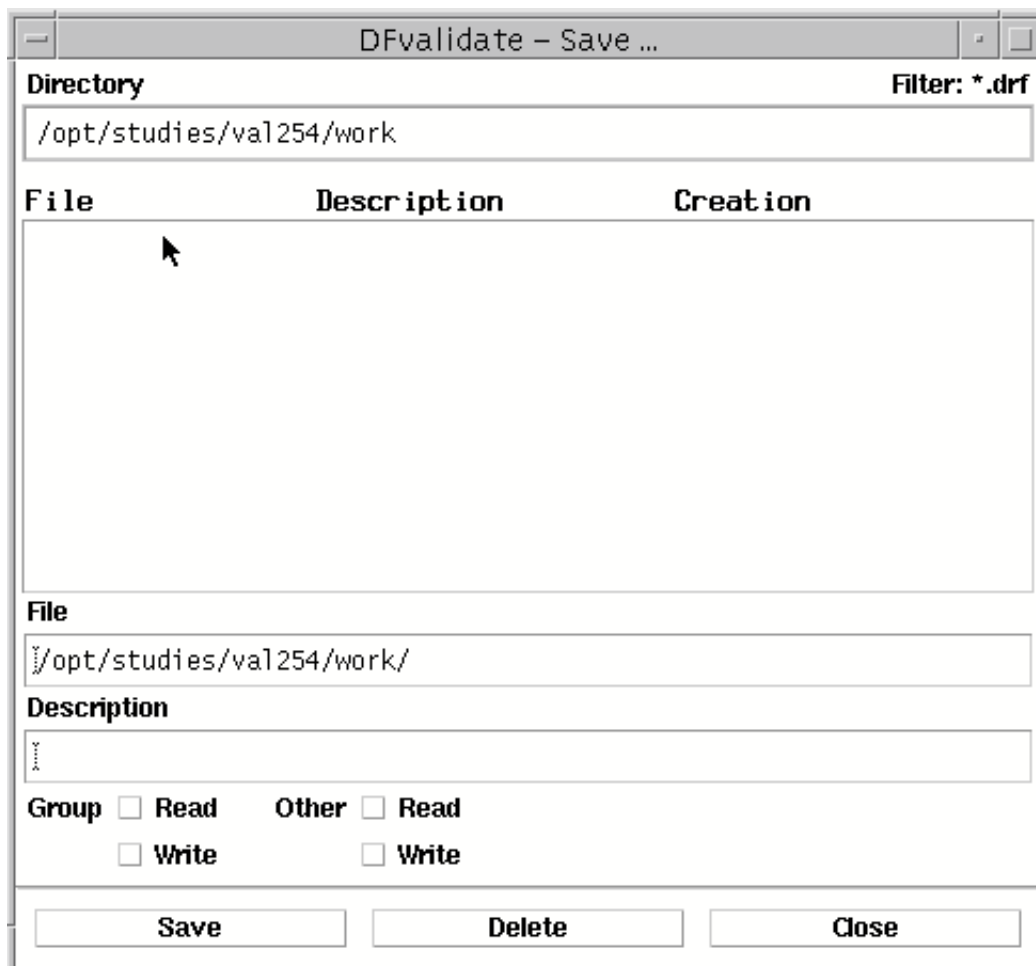
You can save your current record set to a DataFax retrieval file by choosing Save from the Set menu. This will allow you to return to the saved set at any time in the future using the By DataFax Retrieval File retrieval type.

To save your current record set:

1. Choose Save from the Set menu.

The following Save dialog will appear.

Figure 5.41. The Set > Save dialog



2. Navigate the Set > Save file selection dialog to specify a destination file name.

By default, the working directory for the study is used and you will have to specify a new filename or select an existing one. You will be warned if you attempt to overwrite a file that already exists.

3. Click OK.

The current record set is saved to the named file. If this was an existing file, and you do not have write permissions to it, the operation fails and an error message indicating the failure and reason is displayed. If you are not the owner of the file, but have write access, the original owner of the file is retained when the current record set is saved.

To retrieve the records listed in the save file at a future time:

1. Choose Build from the Set menu.

The Build has been described in Retrieving Data Records.

2. Select By DataFax Retrieval File as the retrieval type.

The retrieval specifications window will change.

3. Specify the name of the saved file in the Keys Filename field.

You can either specify an absolute pathname or a pathname relative to the work directory for the study.

5.4.2.8. Save As PDF

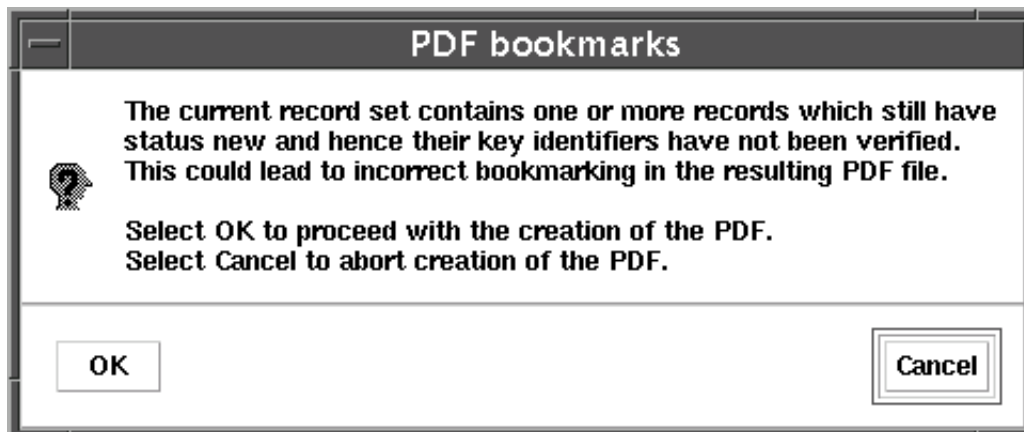
The Save As PDF option allows you to create a bookmarked PDF file of the current set of CRFs where the current set may contain existing, raw data entry, or new records.

To save your current record set to a PDF file:

1. Choose Save As PDF from the Set menu.

The Save As PDF dialog will appear if the set consists of existing or raw data entry records. If the set contains new records, there is a possibility that the key fields used in bookmarking may have been misread by the ICR. Hence, the following question dialog will be displayed.

Figure 5.42. PDF bookmarking dialog for new records

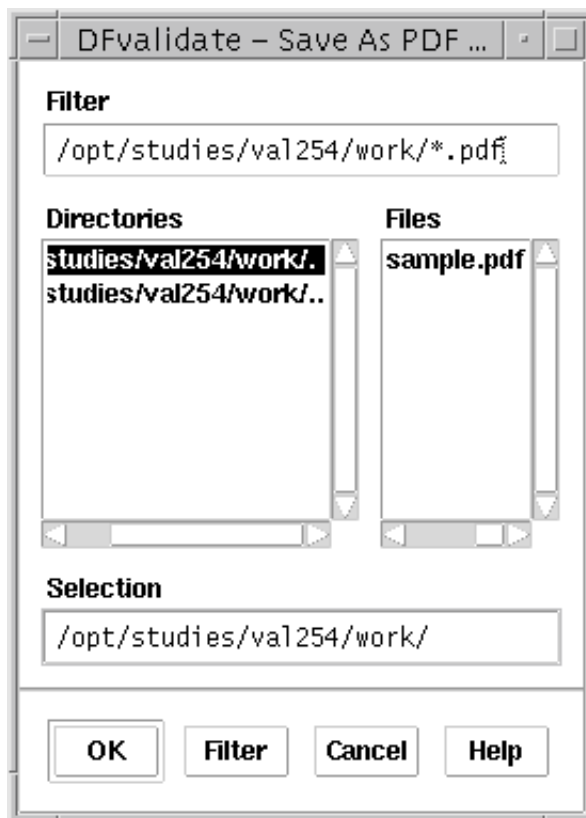


At this point, you have the opportunity to continue with the creation of the (possibly incorrect) bookmarked PDF or cancel the operation. If you choose to continue, the Save As PDF dialog will appear.

2. In the Save As PDF dialog, specify a destination file name.

By default, the `work` directory for the study is used and you will have to specify a new file name or select the existing one. You will not be warned if you attempt to overwrite a file that already exists. Specify a full directory pathname and the name of your destination file in the Selection box.

Figure 5.43. The Save As PDF dialog



3. Click OK.

The current record set is saved to the named file.

To retrieve the PDF file, you must run Acrobat Reader, specifying the name of the PDF file you created.

5.4.3. Record Menu

The Record contains the functions needed to assign record status and to navigate through record sets.

5.4.3.1. Clean

Choose Clean from the Record menu to set the status of the current record to `clean`.

5.4.3.2. Dirty

Choose Dirty from the Record menu to set the status of the current record to `dirty`.

5.4.3.3. Error

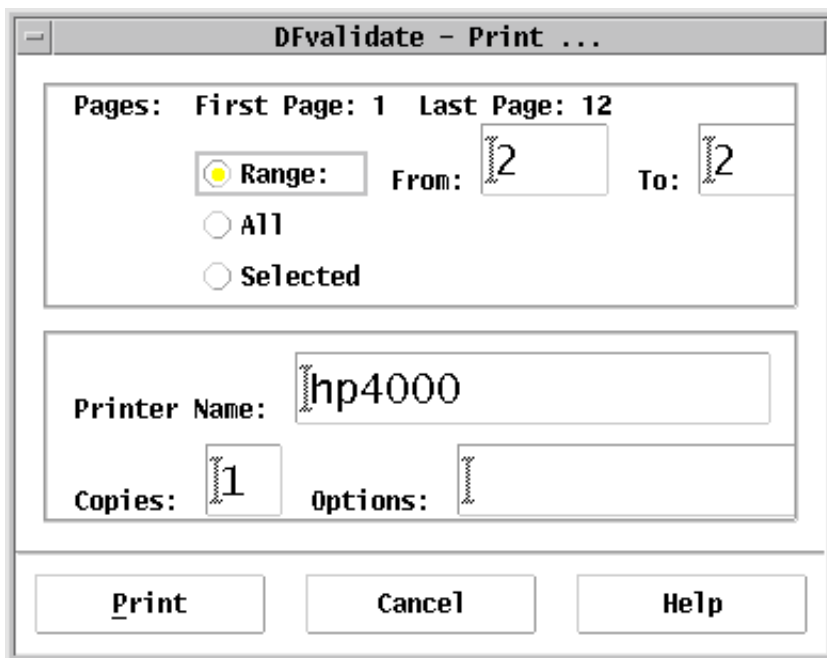
Choose Error from the Record menu to set the status of the current record to `error`.

5.4.3.4. Print

The CRF page images in the current retrieval set can be sent to any printer on the network by selecting Print from the Record menu. This function brings up the following dialog box, which allows you to control which pages are printed, how many copies are made and which printer is used. You may also want to change the font size of the printed text by typing an entry in the options field (see Print Dialog for further information about print settings).

Figure 5.44. The Print dialog

A single copy of page 2, from the current retrieval set of 12 pages, will be sent to a printer named hp4000.



Note

For any retrieval set you can, if you choose, use the Print dialog to print a subset of pages or the entire retrieval set. Before you attempt this, however, you should check with your system administrator to verify that your print spooling daemon has sufficient disk space to accommodate such a large print request. Each CRF page requires approximately 25 Kbytes of disk space for storage and hence attempting to print 100 or 200 pages, for example, may exhaust the disk allocation in the spool directory.

Information for records entered using Raw Data Entry may also be printed. Instead of a fax image you will see ID #, Visit #, Plate #, Image number, record status, and validation level information for each record. Each item will be separated by a | delimiter.

5.4.3.5. Next

Choose Next from the Record menu to advance the current record to the next record in the set. If the current record is the last record in the set, the new current record becomes the first record and **DFvalidate** generates an audible warning beep.

5.4.3.6. Previous

Choose Previous from the Record menu to set the current record to the previous record in the set. If the current record is the first record in the set, the new current record becomes the last record and **DFvalidate** generates an audible warning beep.

5.4.3.7. Top

Choose Top from the Record menu to scroll both the data and the CRF windows to the top of the record. The current variable is set to the first variable (either the visit number or the patient ID number) on the record.

5.4.3.8. Bottom

Choose Bottom from the Record menu to scroll both the data and the CRF windows to the bottom of the record. The current variable is set to the status variable.

5.4.3.9. Reset

Choose Reset from the Record menu to return the data record and QC notes to the last committed version in the database. This can be used to undo changes made to a data record, but it must be done before record status is set. Setting record status commits it to the database.

5.4.3.10. Context

Occasionally you may have difficulty determining where a faxed page came from. In such cases selecting Context from the Record menu will identify the position of the current page within the fax, when the fax arrived, and the phone number of the sending fax machine (provided it was programmed to include this number in the fax). This information is written to the one line message window located between the data and CRF scrolling windows. Here is an example.

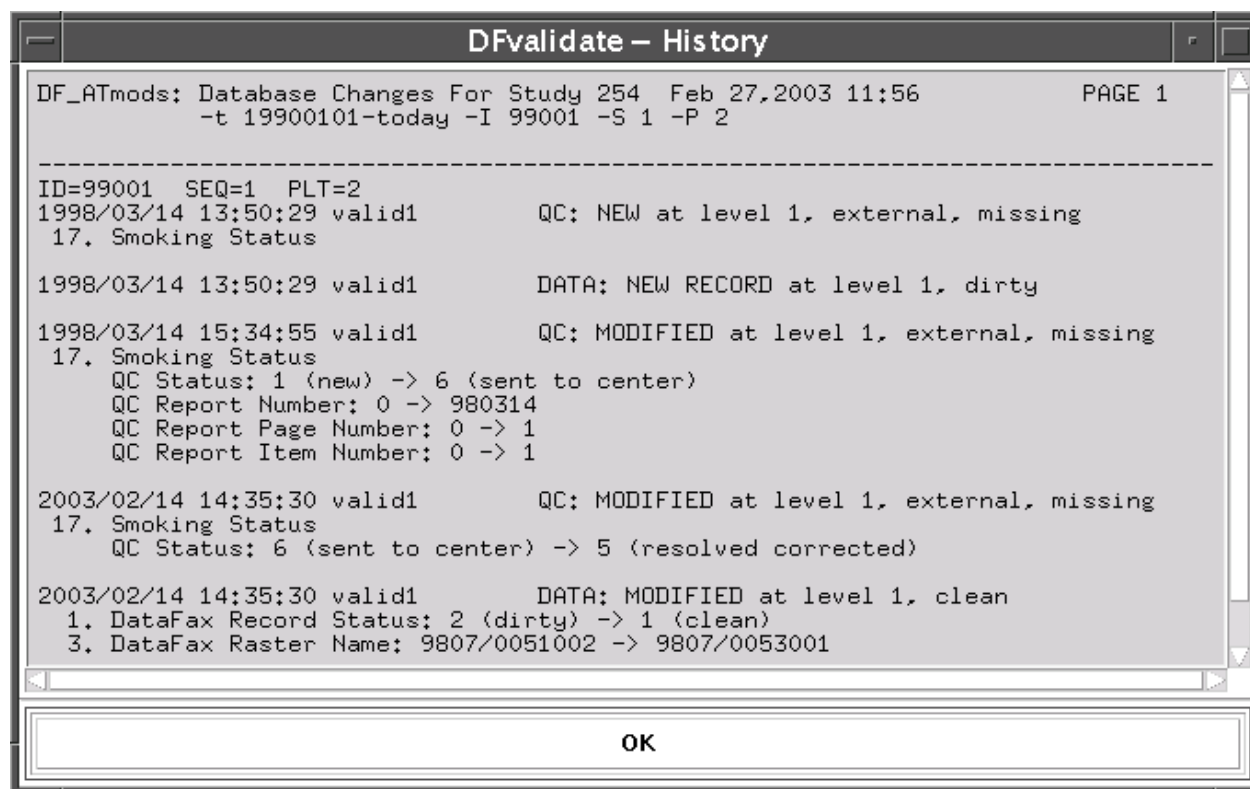
Figure 5.45. A one-line message displayed by choosing Context.



5.4.3.11. History

Selecting History displays the history for the current record. The history is generated by executing the DataFax reports program **DF_ATmods** and displaying the results in a pop-up dialog in **DFvalidate**.

Figure 5.46. History for the current record is displayed in a pop-up dialog.



As the history dialog is a non-blocking window, the user can leave the dialog on-screen and continue with other work in **DFvalidate**. Selecting OK closes the dialog. Whenever History is chosen again, the contents of the dialog, if any, are erased and subsequently replaced by output from the new invocation of History.

5.4.3.12. Delete

You can delete records and the corresponding fax of that page of the CRF by choosing Delete from the Record menu. If you make a mistake you can reset the records status from *delete* to *clean* or *dirty* before releasing the retrieval set. Deletes only become permanent when the retrieval set is released. Watch the status changes in the Record list window to confirm your actions.

You can not delete a primary record which has one or more secondary versions. Attempts to do so will generate a pop-up window which will explain that you must use Dup Resolve from the Misc menu to first select one of the existing secondary records as the primary version. This is necessary to avoid leaving orphaned secondary records with no primary.

Deleting records is not as hazardous as it might at first seem. Remember that you are only deleting the electronic image of the page not the real thing, and that DataFax automatically detects missing pages and puts them on the Fax/Refax List of the quality control reports. So pages which are lost, either inadvertently or on purpose (eg. because of fax noise), will be recovered when the clinical site refaxes the requested pages.

5.4.4. QC Menu

The QC menu allows you to apply, edit, delete and resolve QC notes on specific fields in the database.

5.4.4.1. Add

QC notes are attached to data fields to flag problems on the CRF. Usually these notes are used to ask the clinical site for clarification, but they can also be used simply for internal purposes (eg. so that a data clerk can leave a note regarding something he/she needs clarification on from a data manager or CRA who reviews all QC notes).

A QC note can be added to a field either by choosing Add from the QC menu or by choosing > Add from the fast menu. Either brings up the QC Add dialog box (which we also refer to as the QC note, or electronic Post-it) as shown in the following example. The QC note automatically includes the patient ID, visit, plate, and center to which it belongs, the variable description (from the study database schema), the current value of the data field (if any), an initial guess as to the type of problem, and is flagged for external use without a refax page request (meaning that it will appear in the *Question & Answer List* section of the standard quality control reports). You should modify the Problem, Use and Type options as required. There is also space for a query to be typed if you wish to further explain the problem. However, just selecting a problem type is usually adequate. The variable description, current value, problem type and query (if any) all appear on the QC reports sent to the clinical sites.

Figure 5.47. The QC Add dialog

DFvalidate – Add ...

Id: 1001 (Center: 1) Visit: 1 Plate: 2

Variable

Value

Problem Use

Status Type

Query

Reply

Note

Created

Modified

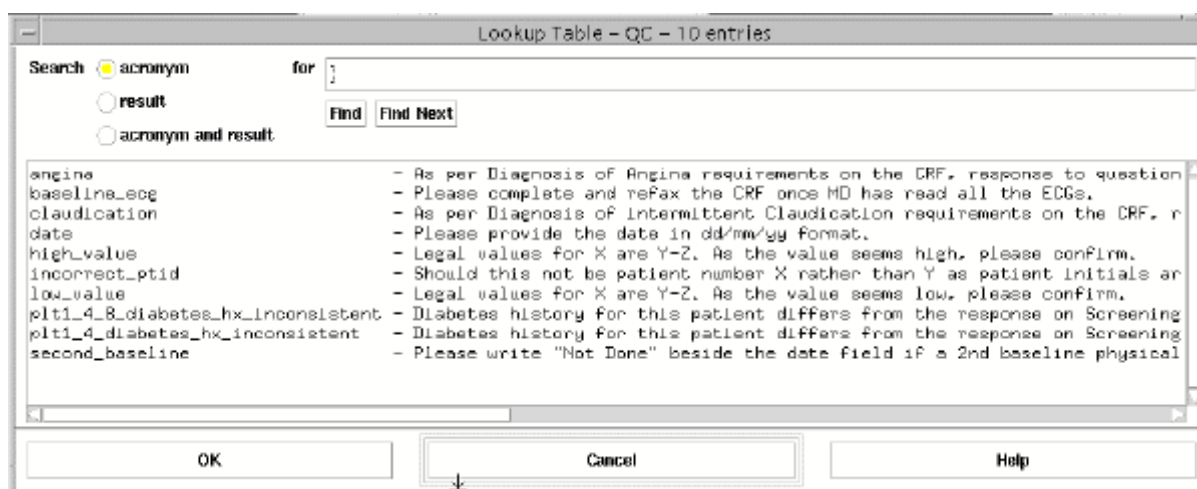
Resolved

DataFax allows you to quickly enter a commonly used query from a pre-defined table (a lookup table) of queries. This feature is just one example of lookup tables, which can be attached to any text field. The definition of QC lookup tables is the same as lookup tables on for text fields, and is described in Lookup Tables. You will know that a lookup table is available by the small ellipsis button to the right of the Query field, as shown in the example above.

To insert a query from the study lookup table, you have three options:

1. **Click on the ellipsis button to display the lookup table in a scrolling list and make your selection from that list.** Double-clicking on a query from the scrolling list will insert the text into the Query value field of the note.

Figure 5.48. A QC Query lookup table used to automate the addition of standard queries in the Validation Tool.



2. **Type the acronym for the query into the Query field and press Ctrl-Enter .** As you can see from the example in option 1, each query has a leading acronym followed by the actual text of the query. The acronyms are an aid for you in remembering and accessing queries from the lookup table. If you can remember the acronym for the query that you want to insert, type it in the Query field and press Ctrl-Enter. If **DFvalidate** can find the acronym uniquely in the lookup table, it will insert the corresponding text (overwriting the acronym you just typed in) into the Query field.
3. **Type the leading part of the acronym into the Query field and press Ctrl-Enter. .** If the acronym or partial acronym that you enter does not uniquely match one entry in the lookup table, **DFvalidate** will display the lookup table scrolling list window and position it so that the first matching acronym is visible and highlighted. Double-click on the entry that you wish to use to insert it into the Query field.

When you have finished adding a QC note select OK to keep it. The QC Add dialog box will disappear and the data field which has the note attached to it will turn blue (the default) to signify that an unresolved QC note is attached. The QC note will be automatically stamped with the login name of the person who created it as well as with the creation date and time.

5.4.4.2. Resolve

When you get a response from the clinical site which resolves a problem identified in a QC report you must both edit the data field to make the correction and mark the problem as resolved in the QC note. Failure to mark the QC note as resolved will lead to the QC note appearing again on a subsequent QC report, something which would quickly come to annoy the people at the clinical sites.

QC notes can be resolved by

- choosing Resolve from the QC menu or by choosing > Resolve from the fast menu.
- enabling the Resolve missing/illegal value QC's as data is corrected preference using the File- > Preferences feature in the validation tool. This preference can only be used to automatically resolve QC notes marked with problem type missing or illegal. The other QC problem types (inconsistent, illegible, fax noise and other) must be resolved manually. (See Preferences)

An example QC note is shown below.

Figure 5.49. The QC Resolve dialog

DFvalidate – Resolve ...

Id: 1001 (Center: 1) Visit: 1 Plate: 2

Variable

Value

Problem **Use**

Status **Type**

Query

Reply

Note

Created samer 06/08/03 13:01:16

Modified samer 06/08/03 13:01:16

Resolved

OK **Cancel**

The resolution status on the QC note can be set to one of 4 values:

1. **Resolved Corrected.** Flag QC notes as corrected if the requested information has been provided. Hopefully, this will be what you use most often.
2. **Resolved NA.** If you are informed by the clinical site that the information you are after was never collected and can not be obtained, resolve the QC note by selecting the not available option. This is different from using the *Data is NA* option described in *Mark Data NA*, although you may often want to use both options together. This is recommended, and is achieved by using *Data is NA* after indicating the QC resolution.
3. **Resolved Irrelevant.** Resolve QC notes with the irrelevant option if it turns out that the information really wasn't needed (eg. you asked for the date of a hospitalization which it turns out never happened).
4. **New.** This option is useful if someone resolves a QC note and then someone else (or the same individual) decides that the response was inadequate and wishes to reissue the note.

If necessary, a brief explanation can also be entered. This becomes part of the QC database and can be used to include documentation notes to explain the resolution (eg. to explain why information is unavailable).

When a QC note is resolved the data field it is attached to changes from blue to magenta (the default colors). Fields with QC notes on them are thus always distinguishable, whether the problem has been resolved or not.

Remember to sign off the record by setting record status to `clean` or `dirty` after you resolve a QC note. The resolved QC note is only written to the QC database when the record is signed off. If you move to another record before setting record status your resolution will be discarded, i.e. the QC note will remain unchanged.

5.4.4.3. Edit

QC notes can be edited by choosing Edit from the QC menu. It is a good idea for a CRA to take responsibility for reviewing all QC notes before they are transmitted to the clinical sites. One approach which works well is to allow data clerks to add QC notes and to select the problem type, usage and refax status and to have a CRA check all notes for accuracy and to add queries if needed. When a QC note is edited the Modified field is stamped with the user name, date and time of the last modification.

Remember to sign off the record by setting record status to `clean` or `dirty` to write the modified QC note to the QC database, before you move to another record in the retrieval set. If you move to another record before setting record status your modifications will be discarded, i.e. the QC note will remain unchanged.



Warning

DataFax updates the Value field of a QC note with the current value of the data field whenever a QC note is edited or its resolution status is reset to New.

5.4.4.4. Delete

Sometimes you might want to delete a QC note. The person who added it may discover a comment written at the bottom of the page which clarified the problem and made the QC note unnecessary, or a CRA or data manager might decide that a QC note added by a data clerk was really unnecessary and decide to remove it. This is accomplished by selecting Delete from the QC menu when either the QC note is open or the data window cursor is positioned at the field containing the QC note.

When a QC note is to be deleted, a warning confirmation dialog appears. Following user confirmation, the QC note is immediately deleted from the database and the field color (previously associated with the QC note) is replaced by the color associated with the current data value (i.e., legal, illegal, optional, or missing).



Important

The QC dialog automatically displays the center number for the patient ID to which the QC note belongs. If a patient is moved to a different center, the system will immediately update all QC notes belonging to the patient with the patient's new center number.

5.4.5. Reason Menu

The actions in the Reason menu permit the user to add a reason to the current value, or a changed value, as well as to approve or reject pending reasons added by **iDataFax** users.

5.4.5.1. Add

A reason may be attached to the current user-defined field by selecting Add and completing the dialog shown in the following figure. It is not possible to attach a reason to a key field, regardless of whether or not the field's value has changed.

Figure 5.50. The Reason Add dialog.

If there is a lookup table associated with the reason dialog, it will appear as an ellipsis button, ..., to the left of the reason text field. This lookup table is used in the same way as those lookup tables for QC notes (See the QC Add description in QC Menu).

The reason may provide optional supporting information for the current value (referred to as the "reason for data value") or provide required documentation for a changed data value (the "reason for data change").

Reasons added in **DFvalidate** are automatically assigned a status of *Approved* - they do not require any additional review or approval. The field is assigned the green color to indicate that it has an approved reason.

There are two scenarios in which a reason is automatically provided by the system:

1. **Key field change.** When a key field (Patient ID, visit/sequence, plate, or study number) has been changed, a reason for change record is automatically inserted into the database with the reason *Keys changed from [a,b,c] to [d,e,f]*, where *a,b,c* are the old keys and *d,e,f* are the new keys. In this situation, the reason for change dialog will not open automatically upon leaving the changed field, and the user will not be prompted to provide a reason for change.

Reason for change records for key field changes always reference the Patient ID field, even if the key change involved the visit/sequence, plate, or study number.

2. **Record deletion.** The Record Delete action displays the reason dialog in addition to the confirmation dialog that is displayed as a result of this action. A reason of `Record deleted by user action` is pre-filled into the Reason field. If the user cancels the dialog, the entire delete operation is also cancelled.



Note

A reason cannot contain control or `']'` characters and can not consist entirely of white space characters.

5.4.5.2. Approve / Reject

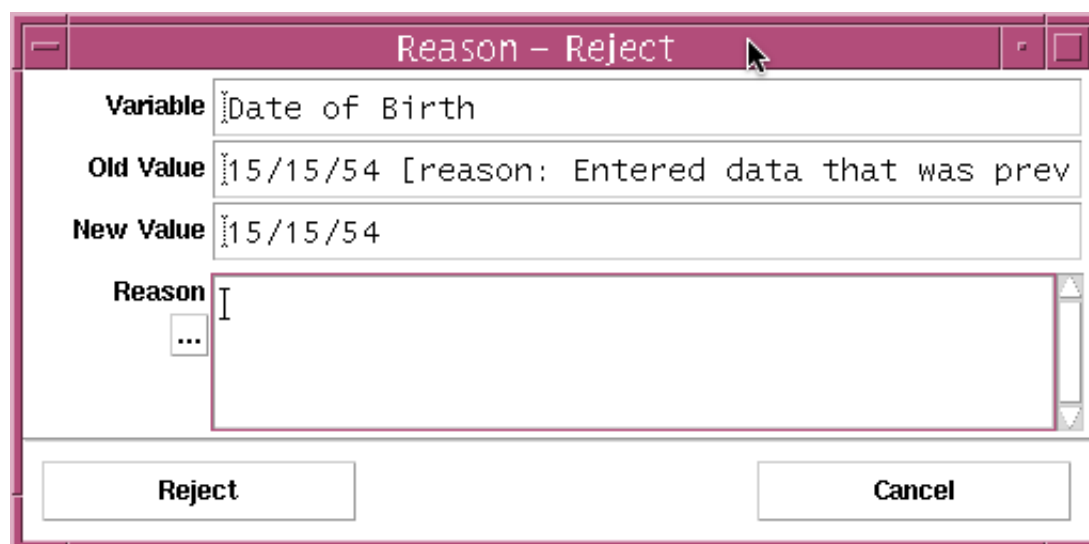
Reasons may also be added by **iDataFax** users to document why a field's data value has been changed, or perhaps to provide additional information regarding an unusual or illegal value. Reasons added by **iDataFax** users are assigned a status of `Pending`.

Reasons with `Pending` status must be reviewed centrally and either approved or rejected. To approve or reject a reason, simply choose Approve or Reject from the Reason menu. No additional input is required.

Figure 5.51. The Reason Approve dialog.

When a reason is approved, the field is assigned the green color.

Figure 5.52. The Reason Reject dialog.



The dialog box titled "Reason - Reject" contains the following fields and controls:

- Variable:** Date of Birth
- Old Value:** 15/15/54 [reason: Entered data that was prev
- New Value:** 15/15/54
- Reason:** A text area with a vertical scrollbar and a small button with three dots (⋮) to its left.
- Buttons:** "Reject" and "Cancel" at the bottom.

When a reason is rejected, it is recommended that the central reviewer add a new QC note to the field explaining why the reason was rejected and possibly requesting additional information.

5.4.6. Options Menu

In addition to the main validation tool window there are five optional windows that you can selectively pop-up or pop-down. These five windows are the Record List, QC Window, Variable Window, Clipboard and the Edit Checks Log window. You can customize which windows appear when **DFvalidate** is started by changing your application defaults (your DataFax system administrator can help you with this), and also change which windows are currently displayed by toggling their visibility setting in the Options menu. The function of each optional window has already been described in Validation Tool Windows.

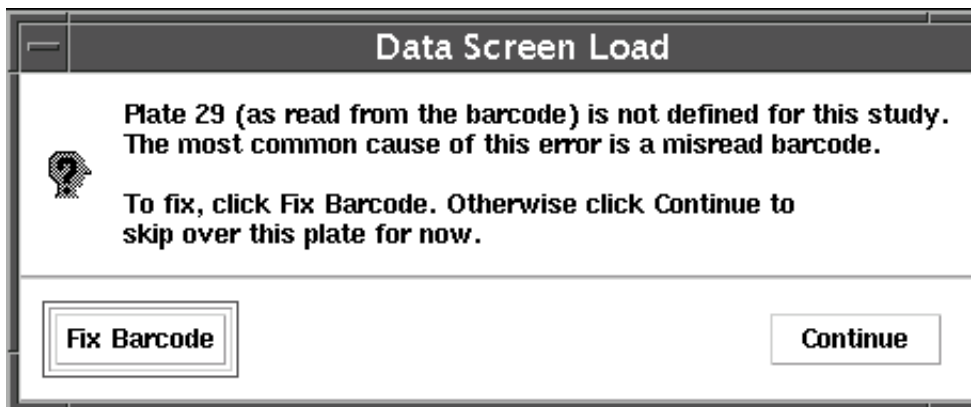
5.4.7. Misc Menu

The Misc menu contains all miscellaneous functions that are not specific to any of the previously described menus.

5.4.7.1. Fix Barcode

DataFax barcodes have been designed for fax transmission and are rarely misread. However, occasionally an error will be made. If DataFax determines that the current plate number is not valid for the current study the following dialog will appear.

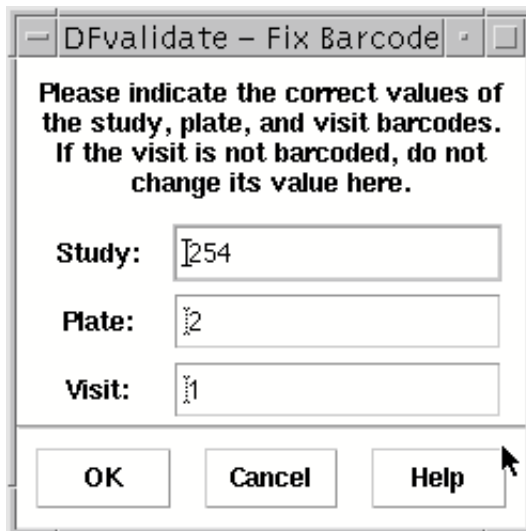
Figure 5.53. The Data Screen Load warning dialog box.



Selecting Fix Barcode in the warning dialog will open the Fix Barcode dialog to allow you to fix the barcode.

If the plate number is valid for the current study, but is nevertheless incorrect for the current CRF page, or if the visit number is in the barcode and it is incorrect for the current CRF page, or if the study number barcode is incorrect, these errors can be corrected by selecting Fix Barcode from the Misc menu, which also brings up the Fix Barcode dialog.

Figure 5.54. The Fix Barcode dialog box.



Enter the correct values and select OK to commit the corrections. Only the keys included in the barcode should be entered. This will always include the DataFax study number and the plate number, but barcoding of the sequence (or visit) number is optional. If the visit number is not barcoded, and instead appears as the first data field on the plate, do not enter it in the dialog. If you are not sure as to the coding of the sequence number, leave it blank and DataFax will warn you if it is expecting one. After changing the barcode one of the following actions will occur:

Table 5.1. Actions following fix barcode.

Change	Action
Study	If the study number is changed the data record is marked for deletion, is removed from the record set, and the CRF image is sent to the unidentified fax router for resolution with DFrouter . If the study number is changed and the record has QC notes attached to it, a dialog will appear warning that all QC notes on the record will be discarded. You will then have the option of cancelling the operation or proceeding. Note that it is not possible to change the study number on a record that has been entered using raw data entry. In this case, you must delete the record and re-enter it in the correct study database.
Plate	If the plate number is changed during new data entry the plate number is corrected, the background image in the data window is updated to correspond to the new plate number, the data fields on the data record are blanked out, the record remains in the new record queue, and the record can be validated and committed to the study database. However, if the data record has a primary or secondary status, it is marked for deletion and removed from the current record set, the CRF image is sent back for ICR processing with the new plate number, and the resulting data record is re-added to the new record queue for the study. If the plate is being deleted from the current set and it has QC notes, these QC notes will also be deleted. In this case a message regarding the deleted QC notes is written to the DataFax error log, but the user is not warned of this deletion.
Visit	When the visit number is changed, the visit/sequence field is updated in the data record and the record and CRF image remain in the current set. The record must be committed to the database by setting record status to save the modification.

The precedence for the above actions is study, then plate, then visit. For instance, if you change the study and plate numbers, only the action for study is taken. In the cases of changing either the study or plate number, a popup will be displayed to warn you that the record is being deleted from the current record set.

Several additional points related to changing the barcode must be kept in mind:

- Plate exit edit checks will not execute when a record is deleted as a result of a Fix Barcode operation.
- If the current record has primary status and there are one or more secondary records, it is not possible to change the barcode values. Changing them in this way would orphan the secondary records. Instead, the status of the current record must be switched to secondary by designating another primary and then the barcode for the secondary can be changed.
- Records deleted as a result of a Fix Barcode operation do not have their corresponding reason for data records deleted. Instead, an additional reason for data record is created which indicates that one or more of the keys (including study number) on the current record has changed. This is useful for identifying the disposition of a CRF, especially if it is the final disposition within this study (because it has been moved to another study).

5.4.7.2. Dup Resolve

Selecting Dup Resolve from the Misc menu will bring up the duplicate resolution dialog box listing all versions (i.e. the primary and any secondary versions) of the current CRF page. As previously described in Resolving Duplicates During Validation of New Records, this dialog box can be used to select which record is to be considered the primary version and to save or delete any secondary versions.

5.4.7.3. Mark Data NA

Sometimes a data field will never be completed because the information is just not available. There are 3 ways in which such fields can be recorded in the database.

- They can be left blank.
- A special code (eg. fill the field with 9s) could be used.
- The field can be flagged as 'not available' by selecting a missing value code from the Mark Data NA pullright menu of the Misc menu.

Data fields flagged NA will receive the NA color code (green). The missing code is written to the database when the record is signed off and will appear in the data field when exported from the database using **DFexport.rpc**. The missing value code does not appear inside the data field in the Validation tool. Instead, when a field is flagged with a missing value code, and subsequently whenever the field has the focus, the missing code and its label appear in the message bar between the split screens.

The Delete key will not remove the NA flag on a data field. To change the status from NA to some other value, you must first select **No Code (reset)** from the Mark Data NA pullright menu of the Misc menu to clear the missing value code, and then enter the desired value. Remember to sign off the record by setting the record status before moving to a new page in the retrieval set to preserve your modifications.

5.5. Recommended Procedures

5.5.1. Validation Levels

- Decide on meaningful validation levels to manage CRF flow through your various data entry and validation steps at the start of the study, and assign these levels to specific staff members.
- At a minimum double validate all pages to reduce database errors.

5.5.2. Retrieval Sets

- Just get one retrieval set at a time unless you really need to have more than one active at a time.
- Release each retrieval set as soon as you are finished with it.
- When validating new records retrieve a whole fax at a time. This will provide you with context information which can be useful (eg. to determine a patient ID# when someone has forgotten to enter it on every page).

5.5.3. Validation Tool Status Window

- For each retrieval set that you create, check to make sure that you are working in the correct mode and at the correct validation level before you begin validating records.
- As you proceed through a retrieval set, validating records, check the updates being made in the status tool window to make sure that the record status and record keys (patient ID, plate and visit numbers) are correct.
- Also check the updates made to the status window, as you move from one record to the next, to confirm that records have been raised to your working validation level. Remember that you must sign off each record by setting record status, even if you have not made any changes to it, to raise it to your working validation level.

5.5.4. QC Notes

- Develop standard operating procedures which data clerks can use to decide when to add QC notes and what problem code to use in different situations.
- Have someone who is familiar with the study protocol and the clinical area review all QC notes before they are sent to the clinical sites in QC reports.
- Remember to check and if possible resolve all QC notes that have migrated to the primary record

after resolving duplicates.

5.5.5. Resolving Duplicates

- Always check the keys (Study, Patient ID, Plate and Visits numbers) to make sure that you really have duplicate records.
- Save old versions as secondary (i.e. do not delete them) at least until you are sure that no mistake has been made in designating the new version as the primary record.

5.5.6. Fax Noise

- If the keys on a faxed CRF are in doubt, either because of fax noise or human error, set the missing key to some unused value and set record status to `error`. Missing records will appear on the refax request list of the standard QC report.

5.5.7. Color

- All users should use the same color map, at least within a given data management group, to aid communication and avoid confusion.

[7] The evaluation of the data field's current value, for the purposes of auto-resolution status and setting the field color, occurs before field exit edit check execution. As a result, if the field exit edit check changes the current value so that it is no longer missing (or no longer illegal), auto-resolution will not occur.

Chapter 6. CRF Viewer

6.1. Introduction

The CRF Viewer (**DFviewer**) is a computerized filing cabinet for study case report forms (CRFs). It can be used to retrieve, view and print the study CRFs which have been received by DataFax. It also allows you to create a PDF file of the CRFs retrieved. The following example shows the main window displaying a CRF (background).

Figure 6.1. The main DFviewer tool window.

The screenshot shows the DFviewer application window titled "DFviewer [valid] DataFax Validation Study". The window has a menu bar with "File", "Actions", "Window", "Split", and "Help". Below the menu bar are buttons for "Previous", "Next", and "CRF List". A progress bar is visible above the form fields.

Form fields include:

- DataFax: #254
- Plate: #001
- Sequence: #000
- Patient Number: 99001
- Patient Initials: ABC (with F, M, L checkboxes below)

BLOOD PRESSURE SCREENING VISITS

Age: 60 years

Sex: ☐ Male, ☒ Female

Race: ☒ Caucasian, ☐ African American, ☐ Asian, ☐ Other, specify: _____

Screen 1: Date 03/03/97 (day/month/year)

Screen 1 Readings:

Reading	systolic	diastolic	mmHg
Reading 1:	170	096	
Reading 2:	170	098	
Reading 3:	172	098	

Screen 2: Date 05/03/97 (day/month/year)

Screen 2 Readings:

Reading	systolic	diastolic	mmHg
Reading 1:	171	096	
Reading 2:	169	096	
Reading 3:	170	096	

Eligibility Criteria

1. Age 16-79 years old. Yes ☒ No ☐

Focus Window: ☒ Top/Left ☐ Right/Bottom

CRF 1 of 24: /opt/studies/val254/pages/9807/0051001

6.2. Retrieving a Set of CRF Pages

There are 4 ways of retrieving CRF pages using the View CRFs By feature:

- by Data Fields
- Program
- DataFax Retrieval File

- Variable.

All retrieval methods are similar to their equivalents in the Validation Tool (**DFvalidate**). Retrieval by Data Fields is the most common.

6.2.1. Retrieval by Data Fields

To retrieve a set of CRF pages, choose View CRFs By from the Actions menu, and enter your selection criteria in the window shown below:

Figure 6.2. The DFviewer retrieval specifications window.

DFviewer – View CRFs By ...

View By **Data Fields**

☐ Sort records within sets by QC note sort order

Center ... 99

Patient Id

Visit ... 1

Plate ... 3

Validation I

Creation Date

Modification Date

Pattern

Status ☒ Clean ☒ Dirty ☐ Error

☐ CLEAN ☐ DIRTY

☐ ERROR ☐ All

OK Apply Cancel Help

CRFs can be assembled by specifying one or more of the following criteria:

1. **Sort records within sets by QC note sort order.** This preference enables/disables sorting of retrieved records in the order specified by the study QC note sort order file `DFqcsort` located in the study `lib` directory. If no `DFqcsort` file is defined, the existing sort order of ID, then visit, then plate (default sort order) is maintained and the preference cannot be enabled. However, if the `DFqcsort` file is present, and the preference is enabled, records are sorted for presentation in the sort order defined by the file. Enabling the preference affects the subsequent sets that are built but has no effect on the current, if any, set. Similarly, disabling the preference affects the subsequent sets that are built

but has no effect on the current, if any, set. To enable the sort preference as the default, change the `app-defaults` file setting for **DFviewer** to read `*.prefQcSort.set: True`.

2. **Center: center number(s) as defined in the study centers database (DFcenters).** When selecting by center, enter a center number, list of centers, or range of centers. If the center specification includes one or more illegal center numbers, an error message is displayed and the retrieval is aborted. If the center specification includes one or more unknown center numbers, the retrieval proceeds, but the unknown center numbers are ignored. An illegal center number is one that falls outside the system-defined range of 0- 2146. An unknown center number is one that falls within the system-defined range of 0-2146, but does not match any of the centers defined in the centers database.

Center and Patient Id criteria cannot be specified in the same retrieval. Doing so generates an error message and the retrieval is cancelled.

3. **Patient Id: patient identification number(s).** Single, multiple and/or ranges of Patient Id numbers can be entered in the space provided.

As above, Center and Patient Id criteria cannot be specified in the same retrieval. Doing so generates an error message and the retrieval is cancelled.

4. **Visit: visit or CRF sequence number(s).** This is the visit or sequence number present as a barcode or data field on every CRF.
5. **Plate: CRF plate number(s).** The plate number as it appears in the second part of the barcode on every CRF page.
6. **Validation.** The current validation level(s) of the corresponding data records (1~7).
7. **Creation Date.** Date(s) on which the CRFs were first validated.
8. **Modification Date.** Date(s) on which the data records were last modified.
9. **Pattern.** A single UNIX style search pattern to be applied to all data records.
10. **Status.** The current status of the corresponding data records (`clean`, `dirty`, etc.).

Multiple retrieval criteria may be specified, in which case only CRFs matching all of the specified criteria will be assembled. Thus in the above example plate 3 for visit 1 will be retrieved for all patients at center 99.

A list of values and ranges may be specified in each field except Pattern. A range is composed of 2 values separated by a dash (-) or a tilde (~). When a list is specified, each value or range (except the last) must be followed immediately by a comma (,). A space may follow commas to improve legibility if desired.

To initiate the retrieval, click OK, or Apply in the View CRFs By window. This initiates a search of the study database for data records that match the specified criteria. The CRFs corresponding to these data records are assembled in a retrieval set which is sorted by Patient Id, Visit, and Plate, and the first CRF page in the set is displayed in the scrolling window, as illustrated in the following example.

Figure 6.3. CRFs are displayed according to the retrieval criteria specified by Data Fields.

DFviewer [valid1] DataFax Validation Study

File Actions Window Split Help

Previous Next CRF List

DataFax #254 Plate #003 Sequence #001

Patient Number 99001 Patient Initials A8C
F M L

MEDICAL HISTORY

	No		Yes		Duration Years & Months	Currently on Treatment		Resolved and/or Controlled	
	No	Yes	No	Yes		No	Yes	No	Yes
1. Hypertension	<input type="checkbox"/>	<input checked="" type="checkbox"/>	→		1 0 6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2. Atrial Fibrillation	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. Other Cardiac EG Arrhythmia	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4. Valvular Heart Disease	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. Myocardial Infarction	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. Kidney Disease	<input type="checkbox"/>	<input checked="" type="checkbox"/>	→		2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
7. Diabetes	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8. Stroke	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9. Transient Ischemic Attacks	<input type="checkbox"/>	<input checked="" type="checkbox"/>	→		1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10. Peripheral Vascular Disease	<input checked="" type="checkbox"/>	<input type="checkbox"/>	→			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Focus Windows: ☒ Top/Left ☐ Right/Bottom

CRF 1 of 3: /spit/studies/val254/pages/9807/0051003

The bottom right corner of this window shows the number of the CRF being displayed, the total number of CRFs in the current set, and the full path name of the CRF image file.



Note

A maximum of 4096 CRFs can be assembled at a time. If more than 4096 records meet the specified selection criteria, only the first 4096 will be assembled and displayed. In such cases add selection criteria (e.g. a center or patient ID range) to narrow the search if you want to review a particular set of CRFs.

Unless you have a very high resolution monitor you will not be able to display an entire CRF page in this window. However, most monitors are wide enough to view the entire width of a standard 8.5 x 11 inch page. The vertical and horizontal scroll bars can be used to adjust the region of the page on display. Once a view region is selected, using the scroll bars, it remains in effect until a new view region is selected. This is particularly helpful when reviewing a particular part of the same CRF page for many patients.

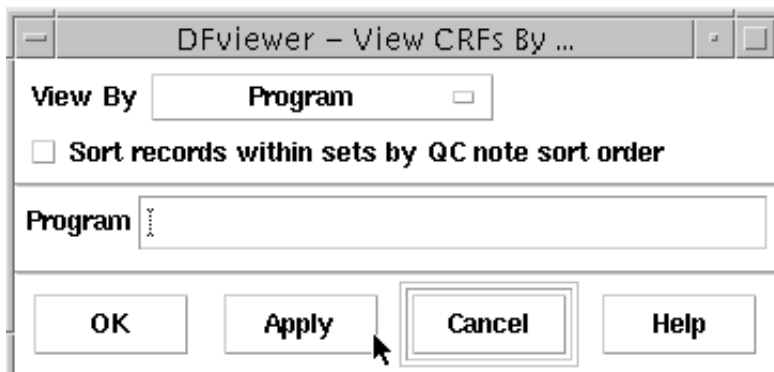


Note

DFviewer can only retrieve CRFs which have been validated to at least level 1 by **DFvalidate**. CRFs which are sitting in the new fax arrival input queue (at level 0 in **DFstatus**) are not accessible from **DFviewer**.

6.2.2. Retrieval by Program

Figure 6.4. The View CRFs By Program specification window.

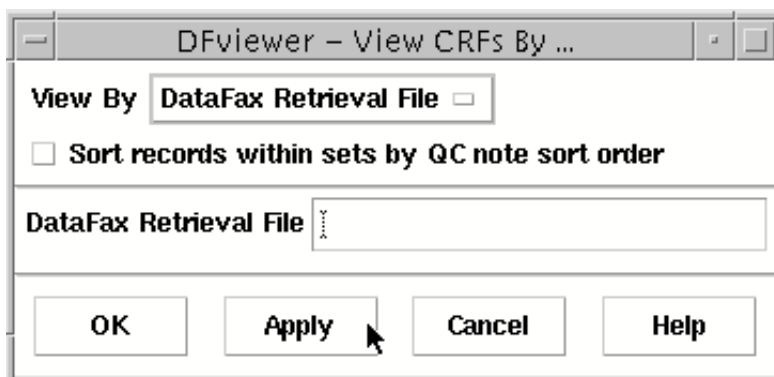


This retrieval method requires a UNIX command as its criteria. You can specify any legal UNIX command line, including argument lists. The requirement is that execution of the program produces properly formatted keys written to standard output.

If a relative name is specified, DataFax assumes that the program (or file) is in the study **DFreports** directory. If a program (or file) is located elsewhere, you must specify the full path name of the program (or file).

6.2.3. Retrieval by DataFax Retrieval File

Figure 6.5. The View CRFs By DataFax Retrieval File specification window.



This retrieval method requires the name of a DataFax retrieval file as its criteria. If a relative file name is specified, DataFax assumes that the file is in the study **work** directory. If the retrieval file is located elsewhere, you must specify the full path name of the file. DataFax retrieval files must always have the extension **.drf** following the file name.

When using DataFax retrieval files, keep in mind the following:

- Records will be assembled in the retrieval set in the same order in which they appear in the file.
- Any non-unique keys will be removed prior to building of the retrieval set.
- All records from the retrieval file, regardless of validation level, are retrieved.

6.2.4. Retrieval by Variable

Retrieval queries built with this retrieval type consist of logical expressions referencing variable names from the study database schema. The current implementation limits you to specifying an expression, referencing variables and values from one plate at a time.

To build an expression, you must first select the plate that is of interest to you. Select the plate from the left column. Selecting a plate updates the middle column with the variable names from that plate. Once you have selected a plate, the steps to building a query follow.

1. Select a variable from the middle list.

Notice that it is inserted into the Query field with some extra punctuation around it. This punctuation is required by the parser in **DFvalidate** - do not edit it. If the selected variable contains any coding (i.e., the variable is a choice or check field), codes and their respective labels will appear in the Codes window near the bottom of the variable retrieval dialog.

2. Select a conditional symbol from the right list.

Most queries involve testing a variable against a value or range of values. To do this a conditional operator is required. If the operator you need is not listed, you can simply type the operator into the Query field.

3. Indicate the conditional value from the keyboard.

If you are testing against a string value, remember to enclose the value in double quotes (").

4. Repeat from step 1 as necessary.

If you are concatenating conditions, you will need to specify a conditional operator such as '&&' or '||' to join the conditions.



Note

You can also enter the query or components of the query directly from the keyboard - the scrolling lists are there as a convenience and a memory aid.

Figure 6.6. Retrieval by variable demonstrates a query that tests for males (`sex == 1`) with a systolic blood pressure of 200 mm Hg or higher (`$(S1SBP1) >= 200`).

DFvalidate - Build ...

Retrieval Task:

Name:

Retrieval Mode: Level:

Retrieval Type:

Query:

Validation:

Plates	Variables	Symbols
1 - Blood Pressure Screening	ID001	(
2 - Patient Entry Form	PINIT001)
3 - Medical History	AGE	== EQUALS
4 - Current Medications	SEX	!= NOT EQUAL
5 - Monthly Follow-up Form	RACE	~ CONTAINS
6 - Patient Evaluation Report	RACEOTH	>= GREATER EQUAL
7 - Study Termination	S1DATE	> GREATER
8 - Medication Log	S1SBP1	<= LESS EQUAL
9 - Adverse Event Report	S1DBP1	< LESS
10 - Death Report	S1SBP2	OR
20 - Date Log	S1DBP2	&& AND
501 - Quality Control Report	S1SBP3	+ ADD
	S1DBP3	- SUBTRACT
	S2DATE	* MULTIPLY

Codes

The constructed query is translated into a UNIX awk script by **DFviewer** and passed on to awk for execution. The result of the execution is a DataFax retrieval file written to the standard output and subsequently read by **DFviewer**.

Retrieval by date variables is a special case of Retrieval by Variables. Retrievals by date variable are performed in the same manner as in **DFvalidate** as described in *DataFax User Guide, Retrieval by Date Variables*.

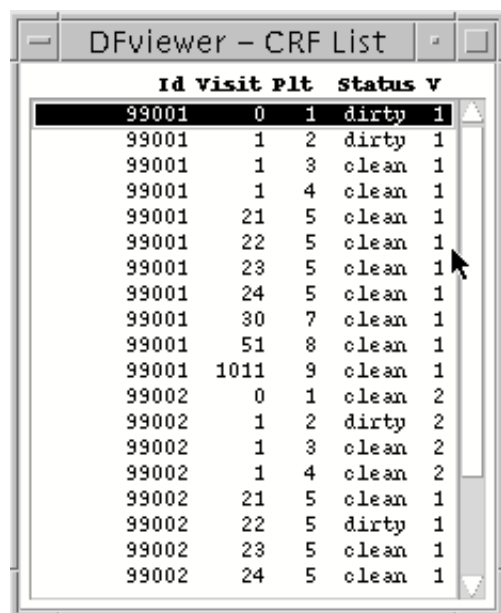
6.3. CRF List Window

In addition to the main CRF display window, the CRF Viewer has 3 other windows which display useful information:

- CRF List window
- CRF Context window
- QCs window

The CRF List window identifies each CRF in the current retrieval set as shown below.

Figure 6.7. The DFviewer tool CRF List window.



The screenshot shows a window titled "DFviewer - CRF List" containing a table with the following data:

Id	visit	Plt	Status	v
99001	0	1	dirty	1
99001	1	2	dirty	1
99001	1	3	clean	1
99001	1	4	clean	1
99001	21	5	clean	1
99001	22	5	clean	1
99001	23	5	clean	1
99001	24	5	clean	1
99001	30	7	clean	1
99001	51	8	clean	1
99001	1011	9	clean	1
99002	0	1	clean	2
99002	1	2	dirty	2
99002	1	3	clean	2
99002	1	4	clean	2
99002	21	5	clean	1
99002	22	5	dirty	1
99002	23	5	clean	1
99002	24	5	clean	1

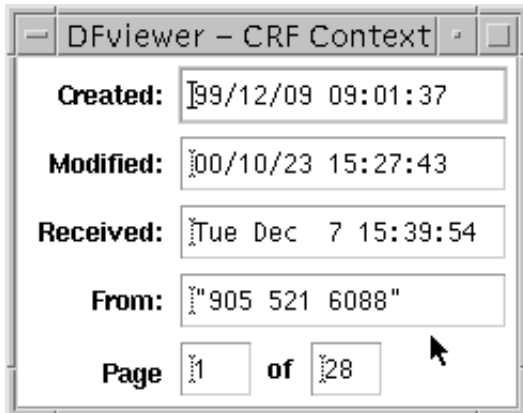
Select the CRF List button or choose CRF List from the Options menu to see the CRF List window. The list identifies the Patient ID, visit, plate, status and validation level of the data record corresponding to each CRF in the current retrieval set. These fields are highlighted for the CRF currently on display. The list is sorted by Patient ID, Visit number and Plate number, and thus its scroll bar can be used to quickly locate records of interest. Clicking on a record in the CRF List will display the corresponding CRF. Also, when the mouse focus is positioned in the CRF List, the arrow keys can be used to display the next or previous page in the list.

When secondary records are retrieved in addition to primary records, records displayed in the CRF List Window are sorted so that for each set of matching keys, the primary record appears first, followed by the secondary records.

6.4. CRF Context Window

The CRF Context window shows where a CRF came from and when it was processed in **DFvalidate**. The CRF Context window is shown below.

Figure 6.8. The DFviewer tool CRF Context window.



The CRF Context window includes:

- **Created.** The date and time the data record was initially validated (typically to level 1) from the new fax queue.
- **Modified.** The date and time the data record was last modified, or validated.



Note

The modification date and time is reset whenever a data record is signed off in **DFvalidate**, even if no data field has been altered.

- **Received.** The date and time the CRF was received by DataFax.
- **From.** The fax identification string (usually a fax number) of the site that transmitted the CRF.
- **Page.** The page number of the CRF within the entire fax.

The first 2 items come for the data record corresponding to the CRF, and the last 3 items are retrieved from the DataFax system-wide fax log. Consequently, if your site has heavy fax traffic this information may take several seconds to retrieve. Also, it is possible that your DataFax system administrator may have truncated the fax log due to space considerations. If so, the last 3 items may no longer be available for all CRF pages in your study.

6.5. The QCs Window

The QCs window displays any QC notes that are attached to the CRF image.

Figure 6.9. The DFviewer tool QCs window.

The screenshot shows a window titled "DFviewer - QCs". It contains several input fields and buttons:

- Variable:** A text box containing "Pulse beats/minute".
- Value:** An empty text box.
- Status:** A text box containing "In sent report".
- Problem:** A dropdown menu showing "Missing".
- Use:** A dropdown menu showing "External".
- Type:** A dropdown menu showing "Refax".
- Query:** A large empty text area.
- Reply:** A text box containing "4".
- Note:** A large empty text area.
- Created:** A label followed by "valid1 98/03/14 14:00:23".
- Modified:** A label followed by "valid1 98/03/14 14:00:23".
- Resolved:** A label.
- Buttons:** "Previous" and "Close" buttons at the bottom.

6.6. Retrieval Sets

Unlike **DFvalidate**, **DFviewer** permits only one set of CRF pages to be retrieved at a time. When subsequent retrievals are performed the current retrieval set is replaced by the new one.

6.7. Record Locking

Unlike **DFvalidate**, **DFviewer** does not lock the CRF pages that you retrieve. **DFvalidate** locks records to prevent multiple users from attempting to simultaneously modify the same data records. But, since data records are neither retrieved nor modified in **DFviewer**, record locking is not necessary. Thus within

DFviewer a CRF page can be retrieved and viewed simultaneously by multiple DataFax users, even when the corresponding data record is locked in **DFvalidate**.

Also, since a user can not modify data records through **DFviewer**, it is a safe tool for individuals who may need to check CRFs from time to time, but who are not allowed to make changes to the database.

6.8. Printing CRFs

To print a set of CRFs first select View CRFs By from the Actions and enter the retrieval criteria needed to assemble the desired CRFs. Then select > Print from the File menu and complete the standard DataFax Print dialog box to identify the pages you want to print.

6.9. Comparing CRFs

You may occasionally find it useful to be able to view two pages on the screen at the same time. Unless you have an exceptionally large monitor it will not be possible to see more than about half a page in each screen, however sometimes this may be exactly what is needed. This can be done by using the Vertically or Horizontally features from the Split menu. In the example shown below, the left window displays all primary records while the right window displays all secondary CRFs for all patients in the database.

To view the CRFs in the following example side by side:

1. Create your first retrieval set (primary records).

Use the Data Fields retrieval option to do this.

2. Select Vertically from the Split menu.

The Focus Window is Top/Left by default so when you retrieve your first CRFs (in this case the primary records) they will appear in the left window.

3. Change the Focus Window to Right/Bottom.

This is done using the simple toggle buttons located in the bottom left corner of the main **DFviewer** window.

4. Retrieve all secondary records.

These will appear in the right window.

Notice that there is a < beside the CRF page that is in the left window and a > beside the CRF that is in the right window. Use the scroll bars on the bottom and side to view other areas on the CRF pages.

Figure 6.10. DFviewer displays primary and secondary records side by side in a vertically split data window.

DFviewer [valid1] DataFax Validation Study

File Actions Window Split Help

Previous Next CRF List

DataFax #254 Plate #002

Patient Number 99002 Patient Initials J F

PATIENT E

Randomization

Provided the patient meets all eligibility criteria, call the random medication code number.

Medication Code #: 1234

Give the patient one bottle of study medication from the

Medical Examination

Date of Birth 05/12/30 day month year

Weight [] [] [] kg or 164

Height [] [] [] cm or 72

Pulse 56 beats/minute

Smoking Status ☐ Never smoked/ ☐ Ex-smoker

CRF 2 of 4: /opt/studies/val254/pages/9807/0048003

DFviewer - CRF List

id	visit	mt	status
98881	1	2	OKREV 3
98882	1	2	OKREV 3
98882	22	6	OKREV 3
98884	1	2	OKREV 3

Focus Window: ☒ Top/Left ☐ Right/Bottom

6.10. Creating PDF Files

The File menu contains the option > Save As PDF which allows you to create a PDF file of the current CRFs. Selecting Save As PDF will save the current set of CRFs to a temporary DataFax Retrieval File (DRF) and this DRF is immediately fed as input to the DataFax **DFpdf** program (see *DataFax Programmer Guide, DFpdf*). The output from **DFpdf** is saved to the filename that you entered in the file selection dialog. The result is a bookmarked PDF file that can then be read by Acrobat Reader. The PDF file contains a bookmarked page and CRF for each record in the current record set.

To save your current record set to a PDF file.

1. Select View CRFs By from the Actions menu.

Enter the retrieval criteria needed to assemble the desired CRFs.

2. Choose Save As PDF from the File menu.

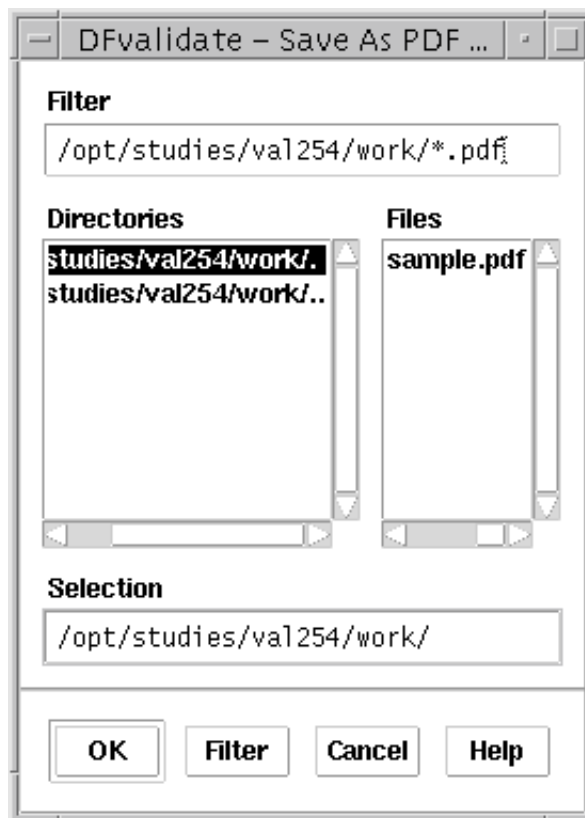
The Save As PDF dialog window will appear as shown below.

3. In the Save As PDF dialog, specify a destination file name.

By default, the `work` directory for the study is used and you will have to specify a new file name or select the existing one. You will not be warned if you attempt to overwrite a file that already exists. Specify the full directory pathname and the name of the destination file in the Selection field. In the

following example, the set of CRFs will be saved to a file called `sample.pdf`.

Figure 6.11. The Save As PDF dialog in which the file `set1` is being saved.



4. Click OK.

The current record set is save to the named file.

To retrieve the PDF file, you must run Acrobat Reader, specifying the name of the PDF file you created.

Chapter 7. Windows-based CRF viewing with DFlite

DFlite is a CRF, data, and QC note viewer. It offers functionality that is similar to the existing CRF viewer in DataFax but enhances it and extends it in several important new directions. Among the significant new features that can be found in **DFlite** are:

- native Windows support - **DFlite** is the first DataFax client program that runs natively in a Windows 98/2000/XP environment, no client-side X Window System support is needed,
- support for wide-area network (WAN) connections where the client/server connection speed is slow,
- multiple study support; **DFlite** is capable of opening simultaneous multiple viewers for multiple studies, and
- the ability to review the stored database data together with the CRF image and the QC notes
- the ability to restrict users to seeing only a subset of the CRF images and their corresponding data records

7.1. Introduction

This document contains step-by-step instructions for most of the tasks that can be performed with **DFlite**.

Read each section before using **DFlite** for the first time. If you do not have the time to read everything before starting, read **DFlite** Basics at a minimum.

If you have previously used **DFlite**, Using **DFlite** is a good reference for the tasks that can be completed with **DFlite**.

7.2. DFlite Basics

This section is a quick tour of **DFlite** and is recommended reading if this is your first time using **DFlite**. These steps provide a quick demonstration of **DFlite** without all of the details.

The quick tour references DataFax study 254, the standard *Acceptance Test Kit* study provided by Clinical DataFax Systems Inc, but any study can be used.

It is assumed that you have an existing DataFax environment and that **DFlite** has been previously installed. If **DFlite** has not yet been installed, refer your DataFax administrator to *DataFax Software Installation Guide*, *DFlite client installation [PK01]*.

1. Start DFlite

The method for starting **DFlite** will depend upon the operating system of the machine from which it is being started.

- **Windows 98/2000/XP.** From the Start menu, pull-right on Programs and select DFlite.
- **Intel/SPARC Solaris.** From a command-line, type:

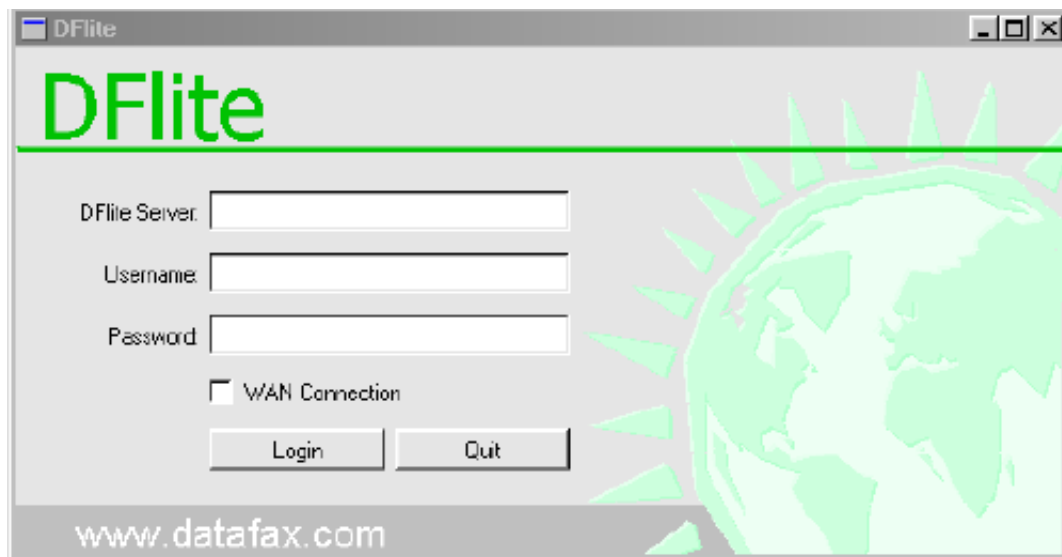
```
% DFlite
```

If your DataFax environment is not DataFax-aware, it may be necessary to type:

```
% $DATAFAX_DIR/bin/DFlite
```

The login screen for **DFlite**, similar to that shown in Figure 7.1, appears.

Figure 7.1. DFlite login screen, Windows 98/2000/XP version



The behavior and appearance of **DFlite** is consistent across all platforms, except for some very minor differences that are related to the windowing controls such as the Close, Minimize, and Maximize buttons.

2. Login

DFlite requires user authentication. Authentication is performed by logging in to the DataFax server and supplying a valid username and password combination. Login by entering the hostname of your DataFax server into the DFlite Server field, and your system username and password into the Username and Password fields respectively. Note that the password is masked as you type it in.

Example 7.1. Completed login screen for DataFax server *dfserver* and username *test*

Leave the WAN Connection check box unchecked at this time. Click Login.



All licenses in use

DataFax is licensed for a maximum number of concurrent users. If the maximum has been reached, login will fail with the message All DataFax licenses in use. Login will not be possible until one of the current users has logged out.

3. Select a study

Following a successful login, the main **DFlite** window appears together with a selection list of available studies. Choose study 254, *Acceptance Test Kit*, from the list, first by highlighting it with a single mousebutton click, followed by OK. The selection list disappears and **DFlite** begins to load the chosen study. It may require a few seconds for **DFlite** to load the study schema - the progress of the load is shown in the status bar at the bottom of the main window.

4. Request CRFs for review

DFlite, like other DataFax tools, uses the term set to describe the collection of CRFs retrieved by a request. To create a set, choose by Data Fields from the Build of the Set menu. The Retrieval By Data Fields dialog, as shown in Figure 7.2, appears.

Figure 7.2. Retrieval By Data Fields dialog

The Retrieval By Data Fields dialog requests a set of CRFs matching the specified retrieval criteria. The default retrieval criteria select all CRFs for `clean` or `dirty` records in the study database.



Request result size

For some studies these default criteria may create a set that is very large. If you are using a study other than 254, you may wish to change the retrieval criteria before proceeding.

Click OK to create a set of CRFs with the specified criteria.

5. Review the requested CRFs

After retrieval, the first CRF of the selected set of CRFs is displayed in the main window.

Example 7.2. DFlite windows after default retrieval

The screenshot shows the DFlite software interface. The main window displays a patient data entry form for 'BLOOD PRESSURE SCREENING'. The form includes fields for Patient Number (99001), Patient Initials (ABC), Age (60 years), Sex (Female), Race (Caucasian), and Screen 1 Date (03/03/97). There are also fields for Reading 1, Reading 2, and Reading 3, each with a 'sys' button. The Record List window on the right shows a table of records for Set 1 - 42 records. The first record is highlighted.

Patient	Visit	Plate	Status	Level
99001	0	1	dirty	1
99001	1	2	dirty	1
99001	1	3	clean	1
99001	1	4	clean	1
99001	21	5	clean	1
99001	22	5	clean	1
99001	23	5	clean	1
99001	24	5	clean	1
99001	30	7	clean	1
99001	40	10	dirty	1
99001	51	8	clean	1
99001	101	9	clean	1
99002	0	1	clean	1

The QC Window on the right contains fields for Variable, Value, Problem, Queue, Status, User, Type, Note, Created, Modified, and Resolved.

The key identifiers for all of the CRFs in the set appear in the Record List window, with the visible CRF highlighted. The entries in the Record List are sorted in the default order, which is increasing numeric order by Patient ID, then Visit, and finally Plate.

Position the cursor in the main window and click the right mousebutton. The main window now displays the current data for the CRF image. Clicking on the right mousebutton again switches back to the CRF image.

Position the cursor over another row of key identifiers in the Record List window and click the left mousebutton. The main window updates to display the CRF image or data for the highlighted row.

Press the down arrow key on the keyboard. The main window and Record List update to display the next record in the set.



Note

The records available to a given user may be restricted if the administrator has implemented access restrictions.

6. Re-sort the record list

Position the cursor over the Visit tab at the top of the Record List and click the left mousebutton. The entries in the record list re-sort by increasing visit number. Click on the Visit tab again. The entries re-sort by decreasing visit number.

7. Split the window into two windows

Choose Split Horizontal from the View menu. The main window is split into two, half-height windows. By default, the top window shows the data and the bottom window shows the CRF image. Of course, the contents of either window can be switched by clicking the right mouse-button while positioning the cursor over the window.

8. Print the current record

Select Print from the File menu. From the submenu, select > Image+Data. Select a printer from the list and click OK. **DFlite** sends the current CRF image and data to the printer for printing in 2-up mode.

9. Exiting **DFlite**

To exit **DFlite**, choose Quit from the File menu.

This section has provided a quick overview of what **DFlite** can do. More detailed information on the capabilities of **DFlite** follows.

7.3. Getting started with **DFlite**

The steps necessary to start **DFlite** are described here for a typical DataFax installation. Your own steps may vary slightly from those described here as a result of local customizations and installation procedures. Be sure to check this with your system administrator before starting.

7.3.1. Pre-requisites

Before starting **DFlite**, the following pre-requisites must be met.

- **Windows 98/2000/XP.** The Windows version of **DFlite** has been installed locally on your computer, or is accessible by you from a shared Windows drive.
- **Intel/SPARC Solaris.** The UNIX version of **DFlite** has been installed in \$DATAFAX_DIR/bin.

Additionally, **DFlite** requires a connection to an operational DataFax installation; this can be either a local network connection, or a wide-area network (WAN) connection, for example via dial-up networking on a telephone line.

7.3.2. Starting **DFlite**

The method for starting **DFlite** will depend upon the operating system of the machine from which it is being started.

- **Windows 98/2000/XP.** From the Start menu, pull-right on Programs and select **DFlite**.

Alternatively, the command to start **DFlite** can be typed at the Start > Run prompt. In a default installation, this command is:

```
C:\Program Files\Clinical DataFax Systems\DFlite\dflite.exe
```

- **Intel/SPARC Solaris.** From a command-line, type:

```
% DFlite
```

If your DataFax environment is not DataFax-aware, it may be necessary to type:

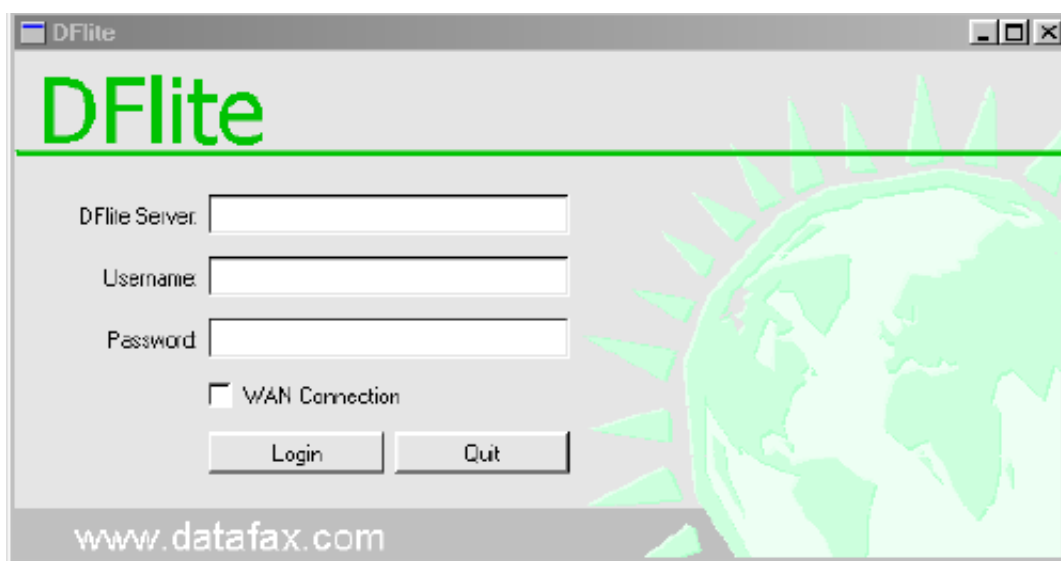
```
% $DATAFAX_DIR/bin/DFlite
```

When **DFlite** has started, the login screen, similar to Figure 7.3, appears.

7.3.3. Logging in

A valid password and username combination is required to access DataFax from **DFlite**. If you are already a DataFax user ^[8], this combination is the same as the username and password that you currently use. Enter the hostname of your DataFax server into the DFlite Server field, and your username and password into the Username and Password fields respectively. Note that your password is masked as you enter it.

Figure 7.3. DFlite login screen, Windows 98/2000/XP version



For security reasons, **DFlite** does not remember your password between sessions. You will need to enter your username and password combination each time you use **DFlite**.

7.3.3.1. WAN Connection

DFlite is able to compensate for a slow network connection if you tell it in advance that the network connection will be slow.

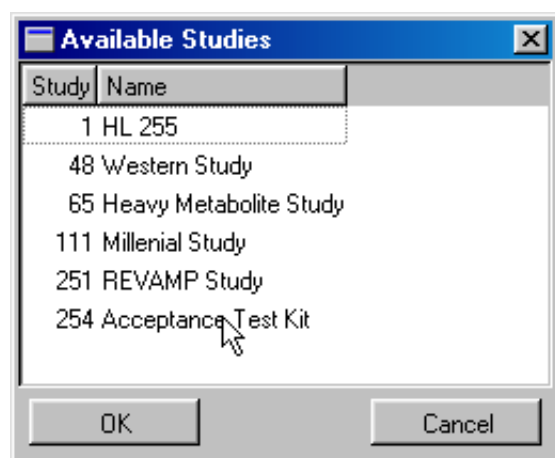
For **DFlite**, a slow network connection is generally defined as any speed less than ISDN, or 128 Kb/sec. Hence any dial-up PPP connection at 28.8 Kb/sec, 33.6 Kb/sec, or even 56 Kb/sec is considered slow. To indicate that you are operating on a slow network connection, check the WAN Connection checkbox. If you are connecting via your local-area network, leave the WAN Connection checkbox unchecked.

Over a WAN connection, **DFlite** uses the idle time while you are reviewing the current CRF to pre-fetch the CRF for the following record. This substantially reduces or eliminates the time spent waiting to flip pages. However, that savings is realized to its fullest when pages are flipped in sequential order. Randomly accessing pages over a WAN connection will diminish the effectiveness of the pre-fetch operation. Additionally, previously viewed CRFs are cached. If you return to a previously reviewed CRF, it can be displayed quickly without requesting it again from the DataFax server.

7.3.4. Selecting a Study

Following a successful login, the main **DFlite** window appears together with the selection list of available studies, as shown in Example 7.3.

Example 7.3. Available Studies list



The list of available studies includes only those studies that you are currently permitted by your DataFax administrator to access. Choose a study from the list by highlighting it with the mouse and then clicking OK, or by double-clicking the study from the list.

Once a study has been selected, the selection list disappears and **DFlite** begins to load the chosen study. It may require a few seconds for **DFlite** to load the study schema; the progress of the load is shown in the status bar at the bottom of the main window.

The functions available for reviewing CRFs in the chosen study are described in the next section.

7.3.4.1. Opening Multiple Studies

DFlite has a multiple study interface in which it is possible to open additional instances of **DFlite** for reviewing other studies. To do this, choose New from the File menu. A new instance of **DFlite** is started together with the selection list of available studies. Choose the desired new study from the list by highlighting it and clicking OK. Each **DFlite** instance can be manipulated independent of the other instances.

To exit an instance of **DFlite** choose Close or Quit from the File menu. Exiting the last instance of **DFlite**, exits the **DFlite** tool completely.

7.4. Using DFlite

This section describes the use and purpose of the various windows of **DFlite**, and the operations that are relevant to record sets.

7.4.1. Tool Windows

7.4.1.1. Main Window

The main window comprises a CRF/Data window, Record List, and QC window.

7.4.1.1.1. CRF/Data window

After building a retrieval set, the main window shows the CRF image of the first record in the set. Depending on the size of your computer screen you may see only a portion of the CRF. Using the vertical and horizontal scroll bars you can adjust the viewable area. To display both the CRF and the data record simultaneously you can split the screen vertically or horizontally by selecting View Split Horizontal or View Split Vertical. By default, this will place the data record in the top (left if split vertically) pane and the CRF image in the bottom (right if split vertically) pane. By clicking the right mousebutton anywhere on the CRF or data record pane you can toggle the view in that pane.

The data view displays the data fields from the database matching the current record and CRF image. Data fields are color coded as to their meaning as shown in Table 7.1.

Table 7.1. Colors used in DFlite and their meaning

Color	Meaning
black	foreground text (fax & data windows) and blank data fields
green	legal/illegal/missing values in data fields
blue	data fields with unresolved QC notes
magenta	data fields with resolved QC notes

Each window can be re-sized horizontally and/or vertically. To resize horizontally, position the cursor over the right-hand edge of the window and drag (the cursor will appear as a double-headed arrow to indicate that re-sizing is possible). To resize vertically, position the cursor over the border separating the two windows and drag.

7.4.1.1.2. Record List

The Record List, an example of which appears in Example 7.4 is a scrollable list displayed in the upper right corner of the **DFlite** window. This list shows the patient ID, visit, plate, record status and current validation level for each record. The set number and total number of records in the set appears in the title above the list.

Example 7.4. Record List window

Set 1 - 161 records					
Patient	Visit	Plate	Status	Level	
17001	0	1	dirty	5	
17001	1	2	clean	2	
17001	1	3	clean	2	
17001	1	4	dirty	5	
17001	21	5	dirty	5	
17001	30	7	dirty	5	
17001	40	10	clean	1	
17001	51	8	clean	1	
26002	0	1	dirty	5	
26002	1	2	dirty	5	
26002	1	3	dirty	5	
26002	1	4	dirty	5	
26002	21	5	clean	1	

The record list window can be closed by selecting Record from the Window menu and deselecting the checkbox. This allows additional vertical space for the QC window or additional horizontal space for the CRF/Data window. The list size can also be adjusted by dragging its border to the desired position.

Clicking on a record in the list will cause that record to become the current one. The CRF/Data view are refreshed to display the current record. The scroll bar or arrow keys may also be used to quickly navigate to records in the list. From the keyboard you can use the arrow keys, page up, page down, home and end keys to navigate through the record list.

7.4.1.1.2.1. Sorting

Records may be sorted in ascending or descending order by patient ID, visit, plate, status or validation level. This is achieved by clicking on the heading of the column you wish to make the primary sort key. The order of sorting on the remaining keys is not affected. Clicking on the same heading again changes to descending sort on the same key, and clicking again returns to ascending order. The order of the columns can also be rearranged by clicking on the column heading and dragging it left or right to the desired position.

7.4.1.1.3. QC Window

The QC window appears in the bottom right corner of the **DFlite** window and displays any QC notes attached to data fields, an example of which is shown in Example 7.5. It displays the Variable Name, Value, Problem, Query, Status, Use, Type, Note, Created date, Modified date, Resolved date attributes of a QC note.

Example 7.5. QC window display for unresolved QC

QC Window	
Variable:	Screen 2 Date
Value:	09/11/
Problem:	Illegal
Query:	Please provide complete date. Year is missing.
Status:	New
Use:	External
Type:	Refax
Note:	
Created:	femy 00/04/04 10:12:47
Modified:	femy 00/04/04 10:12:47
Resolved:	

QC notes can be viewed by clicking, in the CRF or Data view, on any data record field that appears in blue (unresolved QC) or magenta (resolved QC). In addition, the QC window displays the variable name for any data field you click on in either the CRF or data view. The QC window can be closed by selecting Window > QC and deselecting the checkbox.

The QC window can be re-sized horizontally or vertically in the same fashion as previously described for the CRF, Data, and Record List windows.

7.4.2. CRF Operations

The operations that are available with CRFs are described in this section.

7.4.2.1. Retrieving CRFs

To build a set of records for retrieval, select Set Build by Data Fields, specify your criteria and click OK.

CRFs are retrieved by specifying one or more of the criteria:

- Patient ID
- Visit
- Plate
- Validation
- Status

Any criterion that is not specified matches everything. The resulting records are the intersection of all five criterion applied against the database. This is called a set. A range of values may be specified in each criterion by entering the minimum, – or ~, and then the maximum.

Example 7.6. Retrieval criteria for dirty records with validation level 1 on plates 12 through 15 inclusive

Patient Id:	<input type="text"/>
Visit:	<input type="text"/>
Plate:	12-15
Validation:	1
Status <input type="checkbox"/> clean <input checked="" type="checkbox"/> dirty <input type="checkbox"/> error <input type="checkbox"/> CLEAN <input type="checkbox"/> DIRTY <input type="checkbox"/> ERROR	

A list of values is separated by , or one or more spaces. Ranges and lists can be combined as illustrated in Example 7.7.

Example 7.7. Retrieval criteria using ranges, lists, and their combination: records for patients 99001 through 99005 inclusive on visits 0 and 2 through 5 inclusive

Patient Id:	99001-99005
Visit:	0,2-5
Plate:	<input type="text"/>
Validation:	<input type="text"/>

If no retrieval criteria are specified, the default is to retrieve all `clean` and `dirty` records in the study database, including records with patient ID equal to zero and plate 501 (QC reports).

7.4.2.2. Sets of CRFs

Multiple sets may be retrieved, viewed and kept active simultaneously. To switch between sets select Set > Goto to display the Select Set dialog, and choose, with a double-click, the set you wish to view.

Example 7.8. Select Set dialog with 3 sets

The image shows a 'Select Set' dialog box with a title bar containing a minimize button, the text 'Select Set', and a close button (X). The main area of the dialog is a list box containing three entries: 'Set 1 - 23 records', 'Set 2 - 14 records', and 'Set 3 - 89 records'.

An unlimited number of CRFs can be retrieved in a set at one time and multiple sets can be active simultaneously. This is restricted only by the amount of available memory on your computer.

When you are finished reviewing a set of CRFs it is recommended that you release the set, by choosing >Set > Release. This is not required but does free the memory allocated to the set, making it possible to create other sets in low memory situations.

7.4.2.3. Navigating CRFs

Clicking on a record in Record List will cause that record to become the current one. The scroll bar or arrow keys may also be used to quickly locate records in the list. The down or up arrow keys move you forward and backward through the record list one record at a time. You can use the Page Up, Page Down, Home and End keys to navigate through the record list as well. The Page Down or Page Up keys advance/backup by one list height of records at a time. Pressing Home makes current the first record in the list and pressing End makes current the last record in the list.

7.4.2.4. Viewing CRFs

CRFs validated to a minimum level of 1 can be viewed using **DFlite**. By default, the CRF image of the current record appears in the main window. A scrollable split screen window is also available to view both the CRF image and the corresponding data record simultaneously.

The main window can be split vertically or horizontally. By right clicking in either pane you can toggle the view from the CRF image to the data record and back.

It is also possible to view a different CRF or data record in each of the split screen windows. To do this, you must first split the screen by selecting View Split Horizontal or View Split Vertical. By setting Focus to Top/Left, and selecting another record from the record list, the CRF image in the upper or left window changes while the bottom or right window remains unchanged. By setting Focus to Bottom/Right, and selecting another record from the record list, the CRF image in the bottom or right window changes while the top or left window remains unchanged.

7.4.2.5. Printing CRFs

It is possible to print the CRF image, data record, or both from **DFlite**. To do so, choose File Print and then the desired print contents from the choices Image, Data, or Image+Data. A standard print dialog appears. Select the desired print options and click OK.

By specifying a PDF writer as the printer, **DFlite** can create PDF output.

7.5. Limitations

The following limitations are present in **DFlite**.

- CRFs in the new queue are not accessible in **DFlite** until they are validated in **DFvalidate**.
- CRFs that have been assigned a 'pending' status in **iDataFax** are not accessible.
- The record list and QC notes cannot be printed.

7.6. Troubleshooting

This section describes common problems encountered with **DFlite** and their solution.

CRF image doesn't update	If the CRF image doesn't change when you select a different record from the record list, ensure that Focus is set to Both or Top/Left. Choose View Focus and confirm that the checked option is either Both or Top/Left.
Requested record cannot be accessed	The administrator has implemented access restrictions which do not include the keys of the requested record and the current user. Contact the administrator to verify/correct these restrictions if they are being mis-applied.
Cannot print entire record set	Only the record that is currently being viewed prints by default. Choose Print all or a wider Print range to print other or additional records in the set. Printing a very large set may be limited by the amount of physical memory on your computer or printer.
Status bar information is incorrect	The status bar is currently shared by all instances of DFlite . As a result, any update to the status of a DFlite instance will cause a simultaneous update to the status bar of all DFlite instances.
User Authentication failed	One or more of the DFlite Server, Username, or Password values you have entered is incorrect. Please confirm and try again.

[8] If you are not already a DataFax user, your DataFax administrator will need to create a username and assign a password for you, before you can access **DFlite**.

Chapter 8. QC Tool

8.1. Introduction

The QC Tool (**DFqc**) is used to determine the status of all or selected sets of QC notes in the quality control database. For example, it could be used to display and print all unresolved QC notes at a particular center before making a site visit.

8.2. QC Notes

When data management staff identify missing data, illegal values or things they just can't make out due to poor handwriting or fax noise, they flag the problem by adding a QC (quality control) note to the data field. QC notes are coded for:

- Usage (internal/external)
- Refax page request (on/off)
- Problem type (missing value, illegal value, inconsistent, illegible, fax noise, other, missing page, EC missing page, and overdue visit).

QC notes flagged for external use will be included in the next quality control report to be faxed to the clinical sites. Internal quality control notes are not included in QC reports that are sent to study sites. In addition to the QC notes added by data management staff, QC notes which identify missing pages and overdue visits are added to the QC data base automatically whenever the QC Update program is executed. EC missing page QC notes are generated by edit check programs.

8.3. The QC Database

DataFax maintains a data base of all quality control notes (both resolved and unresolved). The status of each note is recorded in the database, and typically progresses from new, not in QC report, to some resolution. The possible status codes are as follows:

- **1 = new, not in QC report.** The status when the QC note is first created.
- **2 = in QC report, not sent.** The status when the QC note is formatted into a QC report.
- **6 = in QC report, sent.** The status when the QC report has been faxed.
- **0 = pending.** The status when a QC note has been replied to but the response is pending approval of data management staff.
- **3 = resolved, not available.** This code is used if the investigator explains that the requested information can not be provided (e.g. some test was not performed).
- **4 = resolved, not relevant.** This resolution code is used if the requested data is not called for (e.g. it appeared that a hospitalization date was missing, but the patient was never hospitalized).
- **5 = resolved, corrected.** Use this resolution code when the requested information is provided.

8.4. Reviewing QC Status

When **DFqc** icon is selected from a DataFax study toolbox the **DFqc** main window appears, with all selection fields blank. With no selection criteria specified you can select Now from the Update to get a total count of all QC notes in the study database by Usage Type, Reply Type, QC Status and Problem Type.

8.4.1. Selecting a Set Of QC Notes

As shown in the following example, this summary can be restricted by specifying selection criteria in any of the fields on the left side of the window, and by selecting any of the category buttons on the right side of the window, before selecting Now from Update. Retrieval task management is identical to **DFvalidate** (see *DataFax User Guide, Retrieval Tasks*). All retrieval methods available for selecting qc notes are similar to their equivalents in the Validation Tool (**DFvalidate**) (see *DataFax User Guide, By Data Fields*).

Figure 8.1. DFqc shows that there are 6 external QC notes on CRFs from centers 10, 20 and 99 for the study named ATK.

DFqc [eric] ATK			
File Update Options			Help
Retrieval Task		Delete	
Name		Add	
Center	...	10,20,99	
Patient ID			
Visit	...		
Plate	...		
Data Field			
Report Date			
Validation Level			
Query Text			
Note Text			
Reply Text			
By:		Date:	
Create			
Last Modify			
Resolve			
<p>Remember that all dates are to be entered in yy/mm/dd format.</p>			
		Usage Type <input checked="" type="checkbox"/> external 6 <input type="checkbox"/> internal 0 Reply Type <input type="checkbox"/> question & answer 0 <input type="checkbox"/> refax CRF page 6 QC Status <input type="checkbox"/> new, not in QC report 0 <input type="checkbox"/> in QC report, not sent 0 <input type="checkbox"/> in QC report, sent 5 <input type="checkbox"/> pending 0 <input type="checkbox"/> resolved, not available 0 <input type="checkbox"/> resolved, not relevant 0 <input type="checkbox"/> resolved, corrected 1 Problem Type <input type="checkbox"/> Missing 3 <input type="checkbox"/> Illegal 0 <input type="checkbox"/> Inconsistent 1 <input type="checkbox"/> Illegible 0 <input type="checkbox"/> Fax noise 0 <input type="checkbox"/> Missing Page 1 <input type="checkbox"/> EC Missing Page 0 <input type="checkbox"/> Overdue Visit 0 <input type="checkbox"/> Other 1	
6 records matched, 0 records ignored			

The ellipses button to the right of the Center, Visit, and Plate fields displays a listing of valid center, visit, and plate numbers, respectively, that are defined for the study. In the situation where a patient has moved from one center to a different center, and you are performing a retrieval by center number, the patient's new center number must be specified in the Center field in order to retrieve all QC notes for that patient, regardless of QC note status.

The Data Field retrieval option allows you to search for a QC note on the specified field number as per the DataFax schema variable numbering defined during setup.

Searches may also be performed on Query, Note, or Reply text by specifying a search pattern in the Query Text, Note Text, or Reply Text fields. The wildcard character '*' may appear in the search string to indicate "contains". For example,

- ***delete***. means that the word 'delete' appears anywhere in the string
- **delete***. means that the word 'delete' appears at the beginning of the string
- ***delete**. means that the word 'delete' appears at the end of the string.

Search patterns are case-sensitive.



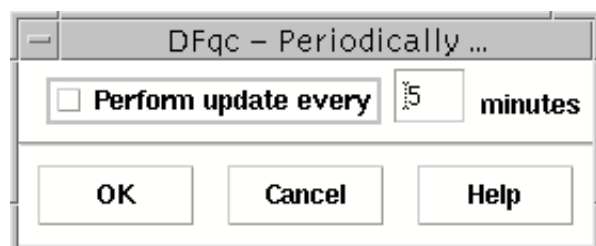
Note

When multiple selection criteria are specified only QC notes meeting all of the selection criteria are counted.

8.4.2. Update - Periodically

To have the QC counts in this window updated automatically at some specified time interval, choose Periodically from the Update. In the dialog that appears, specify the desired interval in minutes, check on the toggle button, and click OK. To subsequently stop periodic updates, pop up the dialog, check off the toggle button, and click OK.

Figure 8.2. The Update Periodically dialog allows you to set periodic updates.



8.4.3. Update - Reset

Choosing Reset from the Update menu will clear all selection criteria and QC counts.

8.4.4. Options - Exact Update and Quick Update

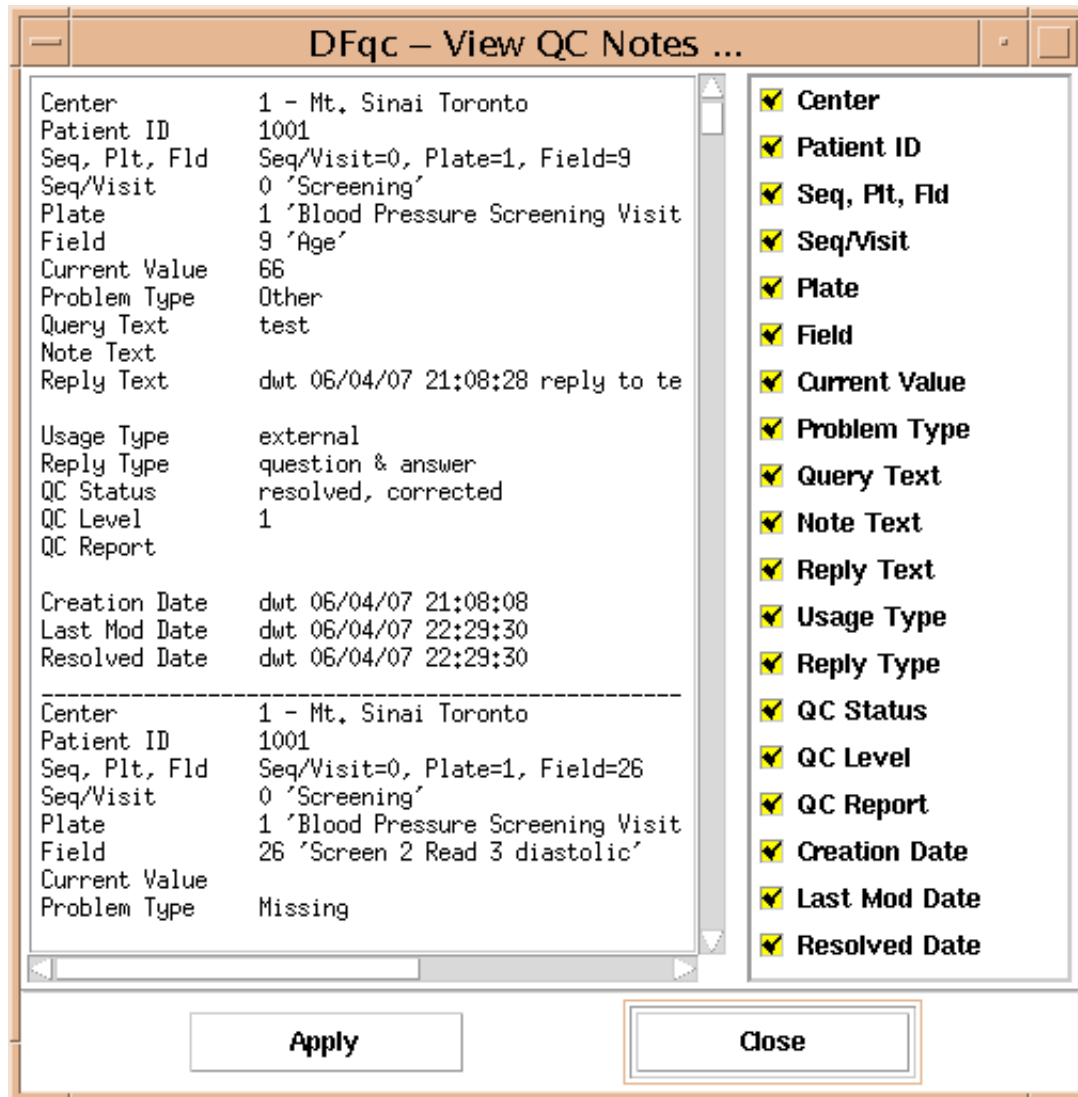
When Exact Update is selected from the Options menu, **DFqc** reads the current QC database each time it updates the displayed counts. In this mode, updates will be accurate even if the QC database is changing because of the work of other users. However, reading the entire QC database takes time. If you are not interested in ongoing changes to the database and simply want to restrict the focus of your search by specifying selection criteria, the response time can be significantly reduced by turning on the Quick

Update option by selecting it from the Options menu.

8.4.5. Options - View QC Notes

After retrieving a set of QC notes you can review them by selecting View QC Notes from the Options menu. Click Apply to display the resulting QC note information.

Figure 8.3. The View QC Notes dialog allows you to view all QC note information if the toggle button for each feature is selected.



In the preceding example all viewing options are turned on. This is the initial default setting. To focus on selected pieces of information from the QC notes, use the toggle button to the left of each feature to select those you want to display, and then click Apply. This will update the View QC Notes window to display only those items which have been selected, as shown in the following example.

Figure 8.4. The View QC Notes dialog allows you to focus on selected QC note information by selecting the toggle button for specific features only.

QC Note 1	
Center	1 - Mt. Sinai Toronto
Patient ID	1001
Seq/Visit	0 'Screening'
Field	9 'Age'
Current Value	66
Problem Type	Other
Query Text	test
Note Text	
QC Status	resolved, corrected
QC Report	

QC Note 2	
Center	1 - Mt. Sinai Toronto
Patient ID	1001
Seq/Visit	0 'Screening'
Field	26 'Screen 2 Read 3 diastolic'
Current Value	
Problem Type	Missing
Query Text	
Note Text	
QC Status	pending
QC Report	

QC Note 3	
Center	1 - Mt. Sinai Toronto
Patient ID	1001
Seq/Visit	1 'Baseline'
Field	7 'Missing Page'
Current Value	
Problem Type	Missing Page
QC Status	
QC Report	

☒ Center
☒ Patient ID
☐ Seq, Plt, Fld
☒ Seq/Visit
☐ Plate
☒ Field
☒ Current Value
☒ Problem Type
☒ Query Text
☒ Note Text
☐ Reply Text
☐ Usage Type
☐ Reply Type
☒ QC Status
☐ QC Level
☒ QC Report
☐ Creation Date
☐ Last Mod Date
☐ Resolved Date

Apply
Close



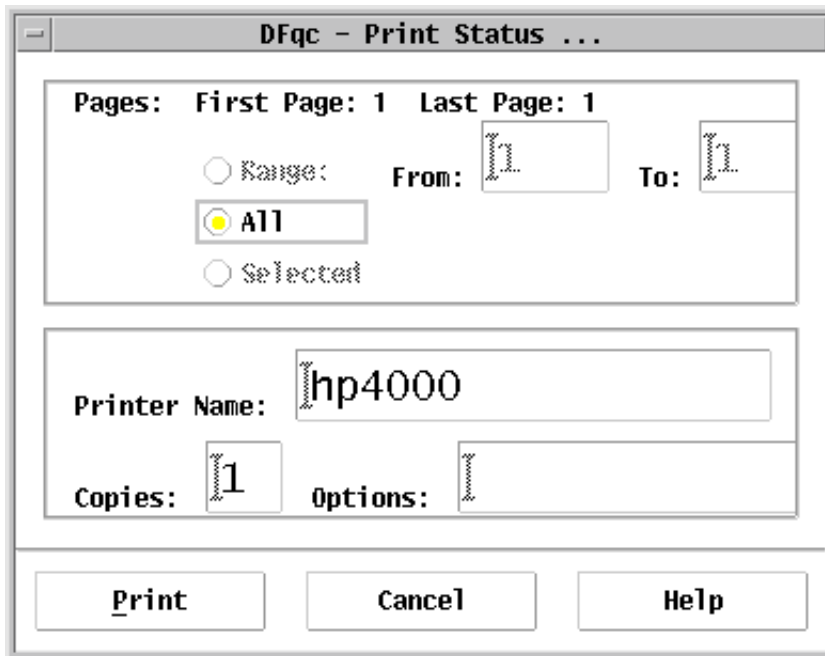
Important

In the situation where a patient has moved from one center to a different center, the View QC Notes dialog will always display the new center number in the Center field. If the QC notes are in a QC report, the QC Report field will display the new center number in the report name in place of the original. The report date, however, will remain unchanged.

8.4.6. File - Print Status and Print Notes

Selecting Print Status from the File menu will print the QC note counts displayed under Usage Type, Reply Type, QC Status and Problem Type. Selecting Print Notes will print all of the QC notes in your current selection set, just as they appear in the View QC Notes window. In either case, printing is controlled using the standard DataFax Print dialog shown below.

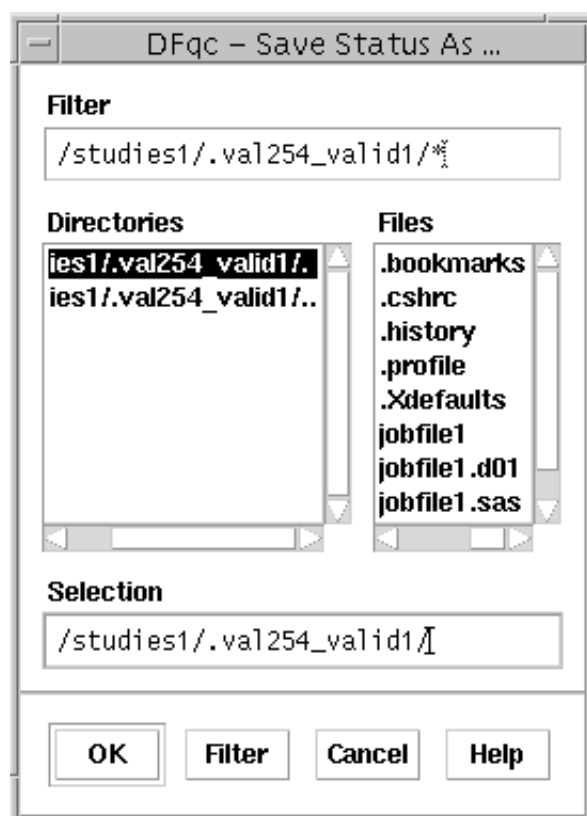
Figure 8.5. The standard DataFax Print dialog used when printing QC note information.



8.4.7. File - Save Status As and Save Notes As

Select Save Status As from the File to write the QC counts displayed under Usage Type, Reply Type, QC Status and Problem Type, to a file on disk, using the standard DataFax Save Status As dialog shown below.

Figure 8.6. The Save Status As is used to specify a filename to which QC note information is to be saved.



Selecting Save Notes As from the File will write the QC notes, just as they appear in the View QC Notes window, to the specified file.

8.4.8. File - Save DataFax Retrieval File

Select Save DataFax Retrieval File from the File menu to create a DataFax retrieval file. This file can be used in **DFvalidate** to retrieve the records to which the current set of QC notes are attached. The current set is saved to the named file. If this was an existing file, and you do not have write permissions to it, the operation fails and an error message indicating the failure and reason is displayed. If you are the owner of the file or if you are not the owner of the file but have write access, you are warned if you attempt to overwrite the file. The original owner of the file is retained when the current record set is saved.

Chapter 9. Lost Data Log Tool

9.1. Introduction

The Lost Data Log (**DFIdl**), is used to register missed assessments and lost CRF pages in the study database.

Occasionally you may learn that a particular patient assessment was not performed and for some reason will never be available. For example, a patient may have missed a scheduled visit, forgotten to keep a patient diary, or refused a lab test. As a result, one or more pages of the study case report forms will be permanently missing. **DFIdl** provides a mechanism for recording these lost pages.

But why is this necessary? One of the DataFax quality control programs (**DF_QCupdate**) which is executed before the creation of quality control reports, detects missing pages and overdue visits and adds QC notes to flag these problems in the quality control database. These QC notes will continue to appear on quality control reports until the missing and overdue CRFs are either received or logged as **lost**. Thus, if an investigator informs you that certain CRFs will never be available, registering them in **DFIdl** will remove them from subsequent quality control reports.

9.2. Creating Lost Data Records

When **DFIdl** is invoked, the window opens blank, with the study name in the frame title, as shown below.

Figure 9.1. The **DFIdl** dialog for the Demo Study.

DFIdl [samer] Demo Study 053

File Edit Actions Help

Patient ID

Visit

Plate

Problem Type

Explanation

Level

Status	Patient	Visit	Plate	Code	Creation Date	Explanation
0 records matched						

To register lost CRFs, you must create a `lost` data record for each CRF page which has been lost.

The steps required to create a single `lost` record are as follows:

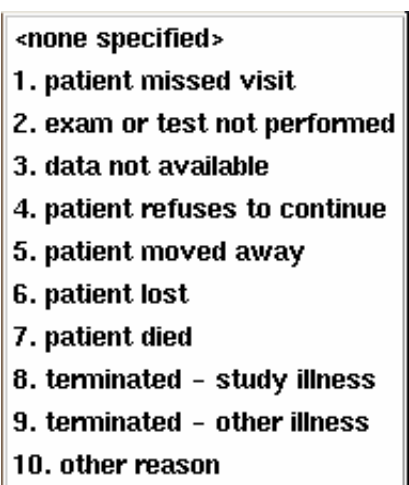
1. Enter the Patient ID, Visit No. and Plate No., corresponding to the page of the CRF which has been lost.

If you select the ellipsis button to the right of Visit No. and/or Plate No. you will get a pull-down menu of all visits and plates defined for the study.

2. Select one of the 10 available reasons from the Problem Type list.

Left click on the Problem Type button to get a pull-down list of the 10 problem types as shown below.

Figure 9.2. The Problem Type pull-down list.



3. Enter an explanation in the text field provided.

The explanation should document the reason why the CRF page was lost.

4. Assign a level.

The pull-down list shows all of the permitted validation levels. Typically, the same validation level as would be used to enter/validate data in **DFvalidate** should be chosen. Choosing a level that is hidden to **iDataFax** users is not recommended as this would prevent them from being able to see, and perhaps replace, the lost records.

5. Select Create from the Actions menu.

The record which you have specified will immediately appear in the scrolling window at the bottom of the **DFIdl** window.

Repeat this procedure to create as many lost data records as necessary. Each new record created will be added to the scrolling list. Create will not create more than one record for each unique combination of Patient, Visit and Plate numbers.

Create does not commit your specifications to the database, it merely collects them in the scrolling window. If at any time you find that you have made an error and wish to start over, select Clear Record List from the Edit menu to clear all `lost` data records from the scrolling window, or select Reset from the Edit menu to both clear the scrolling window and any specifications in the record definition fields at the top

of the window.

The following points should be considered when creating entries in the lost data log.

- **Entering an entire missed visit.** If an entire visit has been missed, it is sufficient to specify the patient and visit number. When Create is selected the program will use the study visit map to determine which plates are required for that visit and will create a `lost` for each of them. Hence, this procedure requires that the visit numbers specified must all be found in the visit map.
- **Entering plates for unscheduled visits.** If plate numbers are supplied when creating an entry, the visit numbers specified are not required to be in the visit map, but simply must be within the legal range for DataFax visit numbers (0-65535). This allows you to create lost records for unscheduled visits.
- **Entering lost records for a range of patients, plates, and/or visits.** If the same reason and explanation apply to a range of patients, plates and/or visits, the entire set may be specified in one step by entering the appropriate range of Patient ID, Plate and/or Visit numbers before clicking Create. When a range of visits is specified together with one or more plate numbers, an entry is made for each possible visit number specified in the inclusive range, regardless of whether or not the visit number is defined in the study visit map.

9.3. Adding Records To DFIdl

The `lost` data records which you have created are committed to the study database by selecting Add from the Actions menu. DataFax first checks the database to make sure that each record is not already in the database. If any of the records displayed in the scrolling window already exist in the database (either as a valid data record indicating that the page was not lost, or as a previously entered `lost` data record) they will be rejected by the Add function, and a pop-up window will display a warning message identifying the number of records which could not be added.

The outcome of the Add operation is displayed under the Status column in the scrolling window, for each `lost` data record. Status will change from `New` to `Added` for each record which was successfully added to the database, and to `Rejected` for any records which could not be entered because they already existed.



Note

Each time that a lost data record is added, DataFax determines if there is an existing missing page, EC missing page, or overdue visit QC note with the same keys, and if there is, deletes it. Conversely, DataFax will not allow the edit check function `dfaddmpqc()` to add an EC missing page QC note to the database if a lost data record with matching keys exists.

9.4. Retrieving Lost Data Records

`lost` data records can be retrieved from the database by entering the Patient ID, Plate No. and Visit No. (or range of values) and then selecting Retrieve from the Actions menu. As in all other DataFax tools leaving one of these fields blank indicates that it is to be ignored in formulating the retrieval. Thus for example, to see all `lost` data records for a particular patient enter the Patient ID number but leave the Plate and Visit fields blank, and then select Retrieve from the Actions menu. All `lost` records for this patient will then appear in the scrolling window. Subsequent retrievals are added to the `lost` records already assembled in the scrolling window.

9.5. Deleting Lost Data Records

To delete `lost` data records you must first retrieve them from the study database. After retrieving a set of `lost` data records, selecting Delete from the Actions menu will remove them from the study database. To delete a record you must have a validation level equal to or greater than the validation level of the person who created the `lost` data record. Again a pop-up window will warn you if any of the records in the scrolling window could not be deleted and the Status column will show the results of the Delete function (Deleted or Rejected) for each record.

9.6. Printing and Saving Lost Data Log Records to a File

Select Print from the File menu to print the current contents of the scrolling window. The standard DataFax print dialog will appear to allow you to specify the printer and number of copies to be printed.

Select Save As from the File menu to write the current contents of the scrolling window to a specified file name, using the standard DataFax Save As dialog box.

9.7. Duplicate Resolution

Occasionally, you may be informed that something will never be available but then it will turn up and be faxed in. DataFax will catch these unexpected arrivals in the same way that it traps duplicate faxes of any page of the CRF. During validation the user will be informed that the new page is a duplicate and that the current record is from the Lost Data Log. The new arrival can then be marked as the primary version of this page and the `lost` data record will be automatically deleted. Alternatively, the `lost` record may remain as the primary record, but this will require deletion of the just received new page and record. It is not possible to have a primary lost record and a secondary data record for the same keys.

A description of duplicate page detection and resolution can be found in *DataFax User Guide, Duplicate Faxes* and *DataFax User Guide, Duplicates and Lost Data Records*.

Chapter 10. Reports Tool

10.1. Introduction

DataFax is distributed with a number of standard report programs which can be used in any DataFax study. Included are reports to:

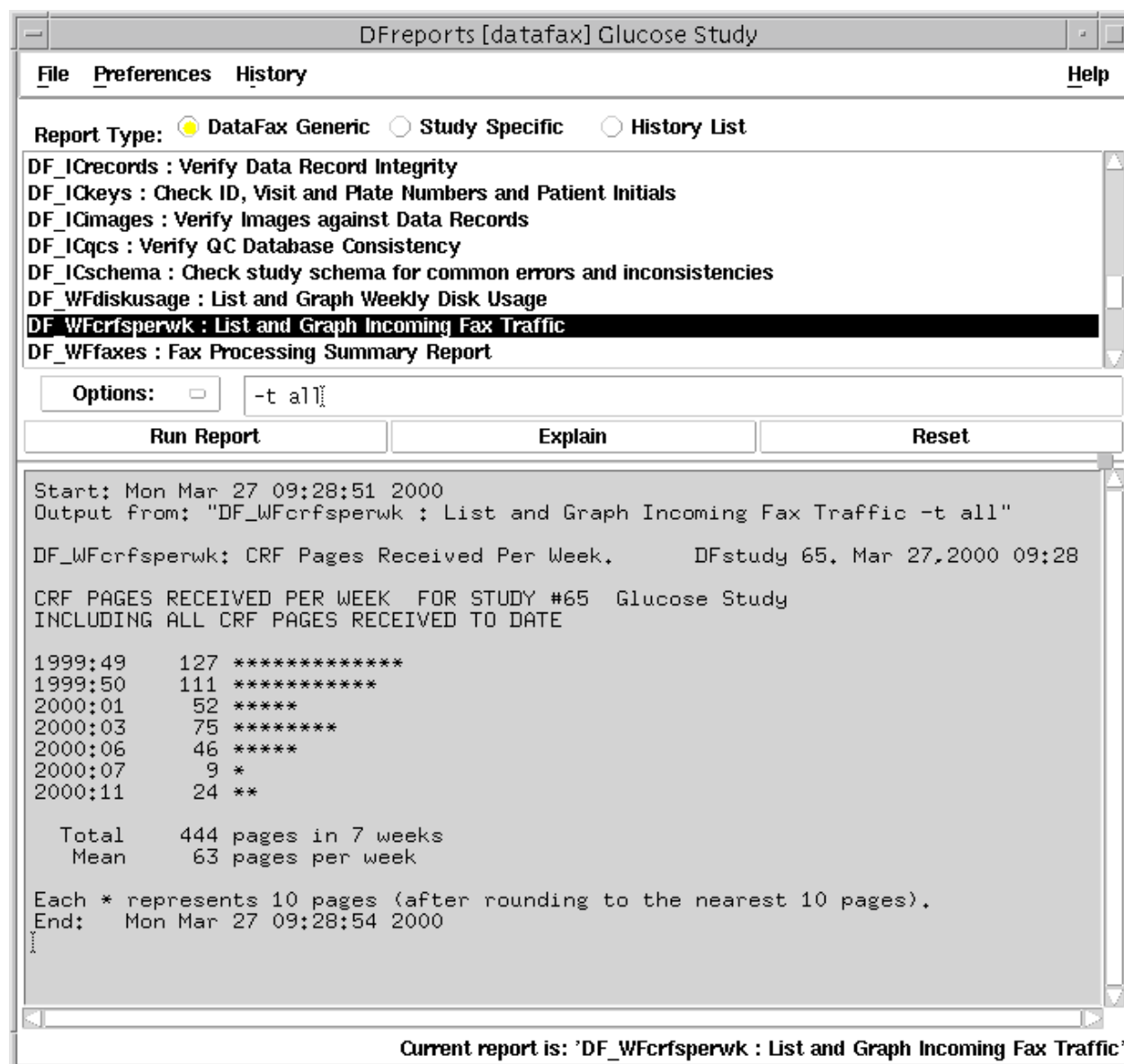
- Review study and center status
- Track individual patients
- Create and fax quality control reports to the participating clinical sites
- Check integrity of the database
- Review work flow

In addition you might create study specific reports, and individual users might want to keep a personal list of report programs which they use frequently. The Reports Tool (**DFreports**) provides a convenient mechanism for managing all such study report programs.

10.2. A Report Example

Clicking the Reports Tool icon opens the **DFreports** window shown below. In this example a standard DataFax program named **DF_WFcrfsperwk** has been executed by first selecting the report name to highlight it, setting the desired option and then clicking the Run Report button. The resulting report is displayed in the scrolling output window.

Figure 10.1. The report **DF_WFcrfsperwk** shows the number of CRF pages received by fax per week from the start of the study to the date on which the report was run, for DataFax study 065.



10.3. Report Types

DFreports maintains 3 separate lists of report programs:

- DataFax Generic
- Study Specific
- History List

Report Type is selected using the button located just above the scrolling list of reports.

Figure 10.2. The scrolling list of DataFax Generic reports.

Report Type: ☒ DataFax Generic ☐ Study Specific ☐ History List

DF_INDEX : Index to Standard DataFax Reports
DF_XXkeys : Export Keys and Visit Dates From Required Plates
DF_XXtime : Update Timing Summary Files
DF_QCupdate : Update QC Database
DF_QCreports : Create QC Reports
DF_QCfax : Fax QC Reports
DF_QCfaxlog : Examine Status of Faxed QC Reports
DF_QCprint : Print QC Reports

10.3.1. DataFax Generic Reports

There are many tasks, common to all clinical trials, which can be performed by a set of standard programs. This reduces the need for study specific programming and promotes consistency in trial management procedures, which in turn increases efficiency and makes it easier to move staff across projects. DataFax is distributed with a number of standard study management and patient tracking reports which you will find useful for all DataFax studies. A list of the current reports is shown on the next page. This list will change as new reports are constantly being added. A list of all DataFax reports available in your current DataFax release can be reviewed by selecting **DF_INDEX**, from the top of the scrolling list of DataFax reports, and then clicking the Run Report button. A complete description of all DataFax Generic reports and their options can be found in *DataFax Standard Reports Guide, Reference Pages*.

10.3.2. Study Specific Reports

Study Specific reports include all executable files which have been placed in the study `reports` directory by local programming staff. These programs might include database integrity checks, a summary of baseline variables by treatment group, or any other program, written in any language, for that specific study.

10.3.3. History List

The History List provides a mechanism for maintaining lists of programs (and program options) which are executed frequently. Each time you start **DFreports** it begins a new history list, to which it writes each Generic or Study Specific program that is executed during that **DFreports** session. You can review the current contents of the list by choosing History List from the Report Type button menu. If this list of programs is one which you know you will be executing often it can be saved by selecting Lists from the History List menu at the top of the screen, and giving it a name by completing the Lists dialog that appears. The next time you need to run these programs the saved history list can be opened from the History List menu.

Each user's reports history list, tagged with the user's login name, is stored in an ASCII text file named `.DFreports.username` in your study `lib` directory. These simple ASCII files can be edited with a UNIX editor (e.g. vi) as required to make changes. There is no limit to the number of history lists that can be kept. Each user of **DFreports** might have several, and thus it might be wise to develop a standard naming convention for all users to follow (e.g. `HLuserInitials_ListName`, as in `HLdwt_StudyStatusReview`).

10.4. Standard DataFax Reports: Synopsis

10.4.1. Audit Trail

- DF_ATcrfs: Track keys associated with a CRF page over time
- DF_ATfaxes: Trace validation history of each page in selected faxes
- DF_ATmods: Trace database modifications over a time period

10.4.2. Center Tracking

- DF_CTcrfs: Summary report of records received from each center
- DF_CTpages: Summary report of CRF pages received by fax and/or email during a specified time period.
- DF_CTqcs: Summary report of external QC notes for each center
- DF_CTvisits: Number of patients who have reached each visit at each center

10.4.3. Integrity Checks

- DF_ICcenters: Verify consistency of the centers database
- DF_ICimages: Verify image references against data records
- DF_ICkeys: Check key fields (id,visit,plate) and patient initials
- DF_ICqcs: Check QC notes for consistency with data records
- DF_ICrecords: Check data records for structural inconsistencies
- DF_ICschema: Check database schema for common errors and inconsistencies
- DF_ICvisitdates: List problems detected by the last execution of DF_XXkeys
- DF_ICvisitmap: Report inconsistencies and errors in the study visit map

10.4.4. Miscellaneous

- DF_icr: Compare original ICR value with current value in data fields
- DF_qcsbyfield: Tabulate and summarize QC notes by problem type
- DF_stats: Display simple variable statistics for a single plate
- DF_XXkeys: Export keys and visit dates from required plates
- DF_XXtime: Prepare timing summary files for other reports

10.4.5. Patient Tracking

- DF_PTcrfs: Display available CRF information for patients
- DF_PTlist: List patient data grouped by patient id and visit number
- DF_PTmissing: Display missing pages and overdue visits
- DF_PTqcs: Summary report of external QC notes for each patient
- DF_PTschedule: Display patient scheduling information from DF_QCreports
- DF_PTunexpected: List unexpected data records found in the study database
- DF_PTvisits: Scheduling and status of all cycles & visits in the visit map

10.4.6. Quality Control

- DF_QCfax: Fax or email QC reports
- DF_QCfaxlog: Examine the transmission date/time and status of faxed/emailed QC reports
- DF_QCprint: Print QC reports
- DF_QCreports: Create QC reports
- DF_QCsent: Mark QC reports as sent

- DF_QCstatus: List existing QC reports
- DF_QCupdate: Update QC database
- DF_QCview: Display requested QC reports

10.4.7. Study Setup

- DF_SScenters: Display study centers information
- DF_SSschema: Display detailed data dictionary information
- DF_SSvars: Display essential variable information from data dictionary
- DF_SSvisitmap: List visit scheduling specifications for a study

10.4.8. Work Flow

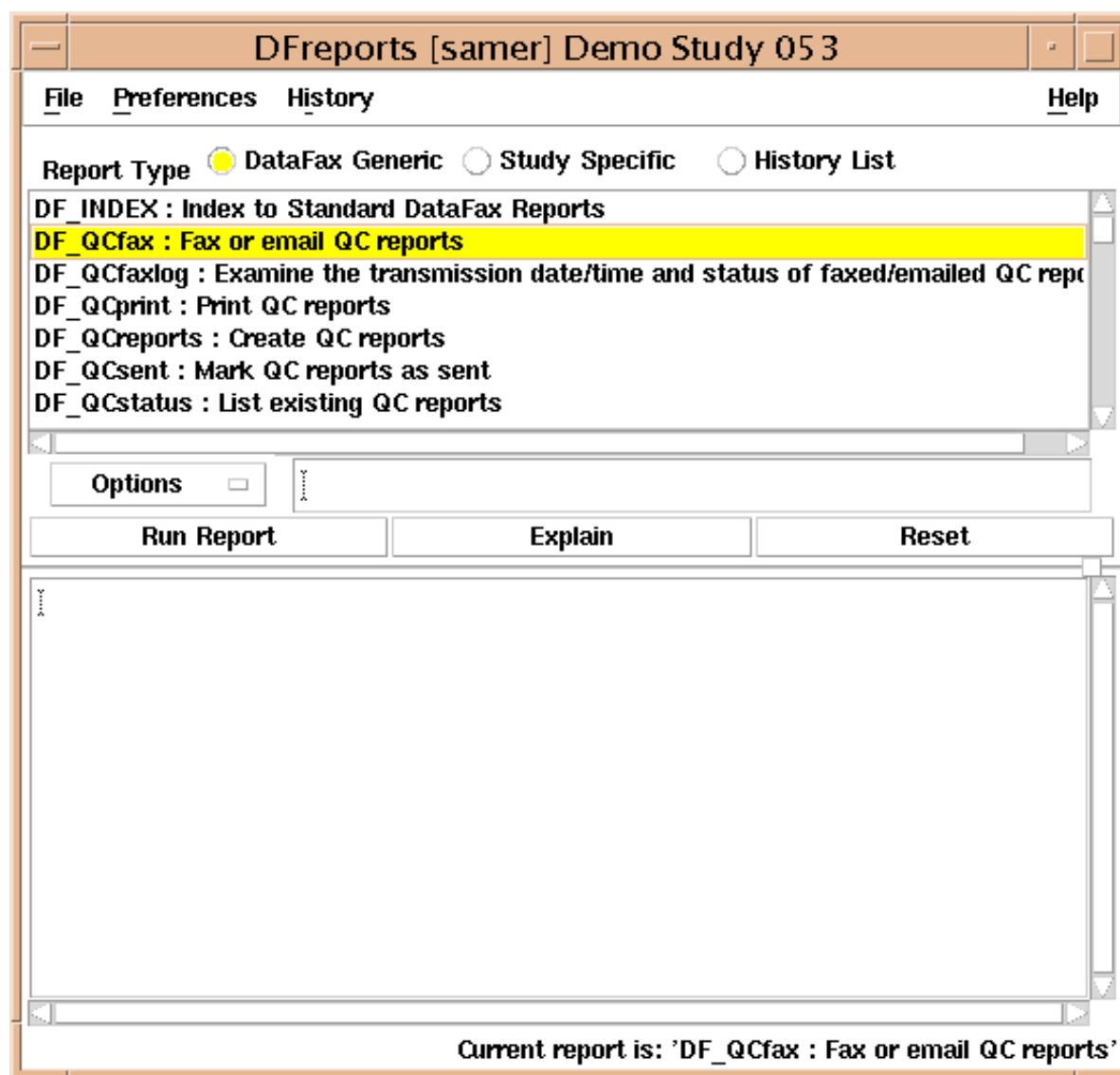
- DF_WFcrfs: Summary of daily journal transactions for CRF validation
- DF_WFcrfsperwk: List and graph incoming fax traffic
- DF_WFdiskusage: List and graph weekly disk usage
- DF_WFfaxes: Summary of elapsed time to 1st validation
- DF_WFqcs: Summary of daily journal transactions for QC notes

10.5. Using DFreports

10.5.1. Selecting a Report

A report is selected from one of the 3 scrolling lists (DataFax Generic , Study Specific, or History List) by clicking on the desired report name. The selected report will be highlighted in the scrolling list, and will also be identified as the Current report at the bottom of the **DFreports** window.

Figure 10.3. DF_QCfax, the program used to schedule the transmission of Quality Control reports back to the clinical sites participating in a trial, is selected as the current report.



10.5.2. Setting Report Options

Most of the standard DataFax reports have optional modes which control some aspect of what the program does when executed. Study specific reports may also be designed with optional modes. After selecting a report, select the Options button to bring up a dialog box showing all of the options available for that report.

Figure 10.4. The options available for the Generic report DF_QCfax.

Clear Options

```

-t ...Test mode.
-f <#> ...Use the specified destination telephone number (only one) for all reports
-F <replyto> ...Use the specified return email address for all reports
-p <password> ...Encrypt the file to be emailed using the supplied password
-P ...Encrypt the file to be emailed using the password specified in the password file ~/.dfpdfpasswd
-r <#> ...Specify the number of retries for each fax to send
-d <#> ...Specify the number of minutes to wait before retrying a failed attempt
-q <#> ...Specify the outgoing fax queue number to be used for sending.
-A4 ...Format the output for A4-sized paper.
-h <hostname> ...Specify the outgoing HylaFAX fax server hostname.
-w <hh:mm> ...Delay faxing until the specified time (hour and minute)
-u <report(s)> ...Fax the specified unsent report(s) (from the QC directory)
-s <report(s)> ...Fax the specified previously sent report(s) (from the QC/sent directory)
-i <name> ...Fax the named internal report(s) (from the QC/internal directory)
-b <yymmdd> ...Re-send a QC report to all destinations to which it has so far failed on all attempts
-a ...Fax all unsent QC reports found in the study QC directory

```

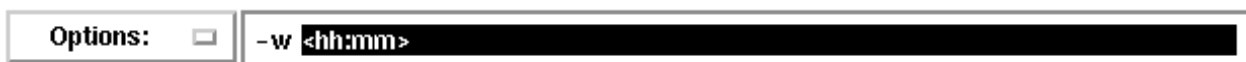
Most options have 3 parts:

- An option flag (preceded by a minus sign)
- A specification to be made by the user (enclosed in <> brackets)
- A brief explanation of what the option does (at the end of each line)

In the above example, this is the case for all but the first option (`-t test mode`), which does not have a specification section, as the `-t` flag is all that is needed to enter test mode.

Although options can be entered in the text window beside the Options button manually using the keyboard, it is also possible to select them, one at a time, from the Options menu, by selecting the desired option line. This is easier and avoids requiring that you remember the syntax for each option. When an option is selected from the Options menu the Options dialog box closes and the selected option appears in the options list, with the specification area highlighted, as shown in the following example.

Figure 10.5. The option selected from the options list for the Generic report DF_QCfax.



For this option the user must then enter the time that the QC report is to be faxed. The highlighted area serves as a reminder of what needs to be entered. In the above example the `hh:mm` is a reminder that time must be specified using a 24-hour clock (e.g. type `23:00` for 11 pm). The entered time will automatically replace the highlighted area. Additional options can be selected, one at a time, from the Options menu in the same way. Each additional option will be appended to the end of the options list. The entire option list can be cleared at any time by selecting Clear Options from the top of the Options menu.

When a new report is selected, the options list is either retained or cleared depending on which mode has been set in the Preferences menu at the top of the **DFreports** window. Usually you will want to clear the options list each time you select a new report because most reports have a unique set of options. However, some reports may be similar in purpose and thus may have similar options. Being able to retain

the options list between reports may sometimes be helpful.

Report options may also be displayed in a usage message within the main **DFreports** output window. A usage message, listing report options and a brief description, can be produced by entering `-u` in the Options text window and selecting Run Report.

10.5.3. Running a Report

Running a report is accomplished by selecting the desired report, setting any desired report options, and then clicking the Run Run Report button. While a report is running, the graphics cursor is displayed as a stop watch. While in this state **DFreports** blocks any attempted input from the user. Currently, no mechanism is provided for aborting a running program. When the program terminates, the graphics cursor returns to an arrow, and **DFreports** once again accepts user input.

10.5.4. The Scrolling Output Window

The output from each report program is written to the scrolling output window at the bottom of **DFreports**. Output will either replace the current contents of the output window, or be appended to it, depending on whether Replace Output, or Add To Output has been selected from the Preferences menu at the top of the **DFreports** window.

When a program starts running, the current date and time, and the report name and options are written to the output window. This is followed by any output generated by the program. When the program terminates the date and time are again written to the output window.

10.5.5. Saving and Printing Report Output

The current contents of the scrolling output window can be printed by selecting Print from the File menu, and completing the standard Print dialog. Also, the current output can be saved to a file on disk in either text or PDF format. Output can be saved to a text file by selecting Save As from the File menu and completing the standard file selection dialog. Alternatively, output can be saved to a PDF file by selecting > Save As PDF from the File menu and specifying a pathname and filename in the resulting dialog.

Also, note that the output window is editable. This allows you to cut and paste sections of the output window into other windows (e.g. xterms or word processing documents), or to modify or delete sections of the output window before it is printed or saved.

10.5.6. Resetting the Reports Tool

Clicking the Reset button clears the scrolling output window and the options list, and un-selects the current report.

10.5.7. Running reports from the command line

DataFax reports may also be run from the command line. To do this, enter the following command at the command line prompt:

```
% $DATAFAX_DIR/reports/reportname [#] [options]
```

where `$DATAFAX_DIR` is the DataFax installation directory, `reportname` is the name of the report, `#` is the required study number, and `options` is the set of report options that one would normally specify in the reports tool itself.

Example 10.1. From the command line, execute **DF_QCreports** for center number 12 of DataFax study number 254

```
% $DATAFAX_DIR/reports/DF_QCreports 254 -c 12
```

If the user simply wants to produce a usage message for any DataFax Generic Report, the following command can be entered at the command line prompt:

```
% $DATAFAX_DIR/reports/reportname -u
```

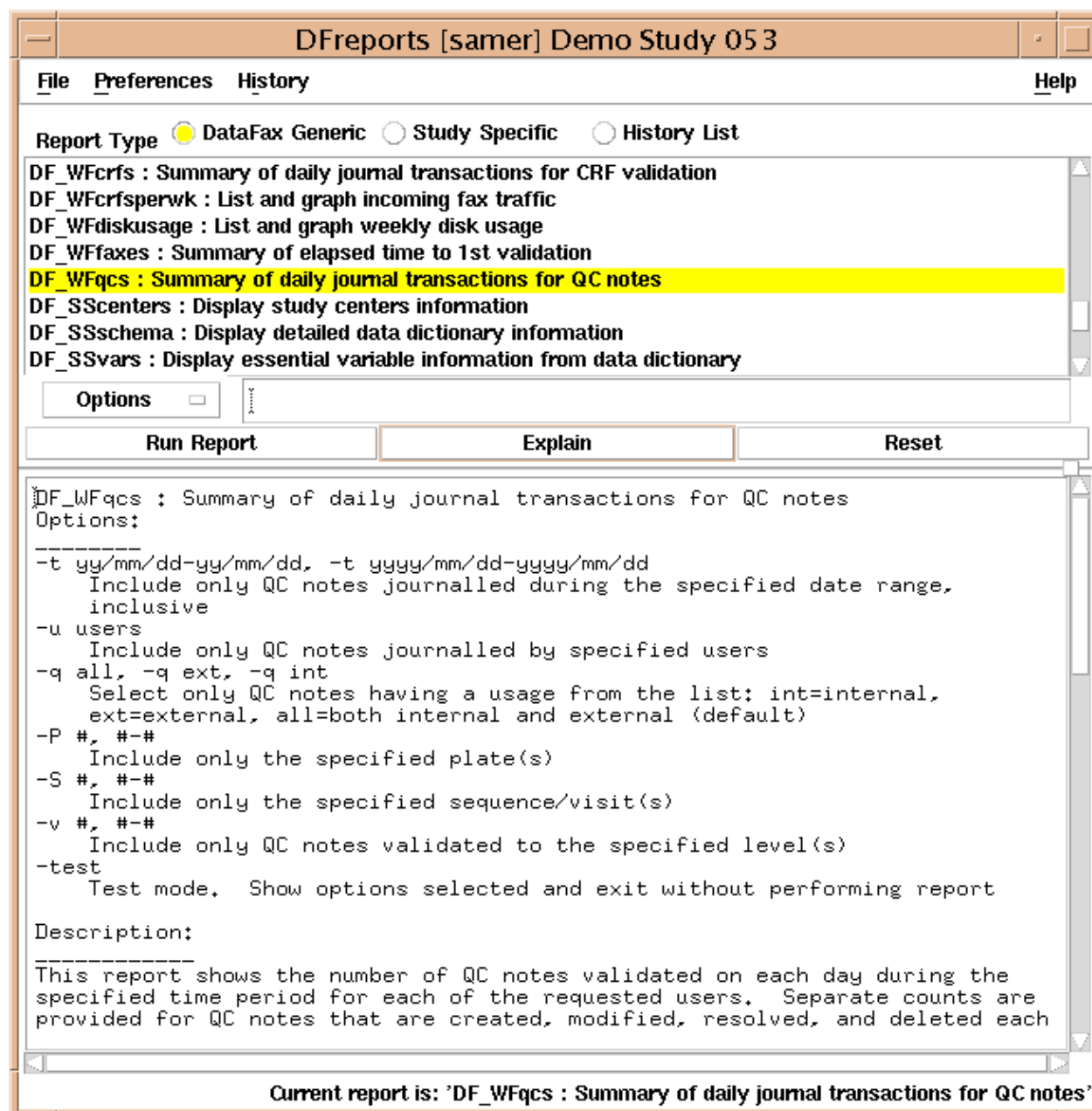
For this command only, the study number argument is not required.

10.5.8. Reading On-line Documentation for Reports

An explanation of each report, including run time options, is available on-line for all of the generic DataFax report programs. After selecting a report click the Explain button to display its on-line documentation.

The following example shows the documentation available for **DF_WFqcs**. All documentation for DataFax reports follows the same format. As shown below this consists of a line showing the program name and options, followed by a brief description of the purpose of the program, followed by a description of each option and several examples, and finally a detailed description of the program.

Figure 10.6. On-line documentation for the DataFax Generic report **DF_WFqcs** displayed by selecting the Explain button.



10.5.9. Creating On-line Documentation for Study Specific Reports

The documentation for all standard DataFax reports is kept in file `.info` in the DataFax `reports` directory, and is updated for each new DataFax release. Consequently, you should not edit this file. However, you may want to examine it as an example when designing on-line documentation for your own study specific reports. The documentation you create for study specific reports must be kept in a separate file, named `.info` in the study `reports` directory

If you have a number of standard programs that are used in all of your DataFax studies, it would be a good idea to keep these programs, and their `.info` documentation file, in a separate directory from which they can be copied to each new DataFax study as part of your standard study setup procedures. Documentation for additional study specific programs can then be appended to the `.info` file in each study `reports` directory.

The general outline and an example of the format for on-line documentation can be found in *DataFax Programmer Guide, Writing Your Own Reports*.

10.6. DataFax Quality Control Reports

10.6.1. Introduction

One of the major DataFax design objectives was to create an automated quality control system, which would include the ability to flag problems seen on the CRFs faxed from the clinical centers, fax back a problem report requesting CRF corrections, and get the corrected CRFs into the database when they were re-faxed. The chapter on the Validation Tool, (DFvalidate) describes how study CRFs are reviewed and validated, how missing data and other problems are flagged with QC notes, and how a re-faxed CRF is detected and validated to become the corrected, primary database record. We will now describe how QC notes are formatted into quality control reports and transmitted back to each of the clinical centers participating in the study. But first we will look at what a DataFax QC report looks like.

Each DataFax QC report, which is faxed back to a clinical center, deals with the patients and problems seen at that center, and has 3 parts which are described in the following 3 sections.

10.6.1.1. Part 1: Patient Status Summary

The first part of a standard DataFax QC report lists all patients being followed at the center. It includes each patient's enrollment date, last follow-up date, and the target date for the next scheduled follow-up. The specific quality control notes are listed in the next 2 parts (Fax/Refax List and Question & Answer List) of the QC report.

As an example, imagine a study in which:

1. Each patient is seen 6 times at monthly intervals.
2. CRF sequence numbers correspond to clinic visit numbers 1 through 6.
3. A special sequence number, 9, is used on a form submitted for any patient who fails to complete the study protocol and terminates early, for whatever reason.

Figure 10.7. The Patient Status Summary of a DataFax Quality Control report.

PATIENT STATUS SUMMARY (* identifies patients with data queries in this report)

PATIENT	ENTRY VISIT	LAST FOLLOW-UP	NEXT FOLLOW-UP
125001	1: 94/01/01	6: 94/06/05	done
125002	1: 94/02/15	5: 94/07/12	6: 94/08/15
125003*	1: 94/02/22	9: 94/05/19	done
125004*	1: 94/03/07	3: 94/06/03	4: 94/07/07

The above figure shows the status of the first 4 patients 125001 to 125004) entered from center #125 for this hypothetical study. In this example, dates are recorded in year/month/day format. Dates will appear on your QC report in whatever date format was used on the study CRFs.

The first patient (125001) entered the trial on January 1, 1994 (94/01/01 and had their last follow-up (visit number 6) on June 5, 1994. All pages of the CRF have been received and reviewed and there are no outstanding problems.

The second patient (125002) also has no problems and is coming due for his final follow-up on August 15, 1994.

The final follow-up received for patient 125003 on May 19, 1994 was reported on a CRF page having sequence number 9, indicating that this was an early termination.

The last patient, 125004, was due for visit 4 on July 7, 1994. Both 125003 and 125004 have outstanding problems with the case report forms received to date, as indicated by the * following the patient number. The specific problems will appear in parts 2 and 3 of the QC report.

10.6.1.2. Part 2: Fax/Refax List

The second part of a standard DataFax QC report lists all CRF pages which have QC notes marked with a refax request, plus any missing pages and overdue visits.

When a CRF page is corrected and re-faxed into the DataFax system, it is recognized by **DFvalidate** as a duplicate of an existing page. The data management staff are then able to confirm that it is indeed a re-faxed page and resolve any QC notes which relate to problems which have been fixed, or issue new QC notes, if necessary, to seek further clarification. Both old and new versions of a re-faxed CRF page may be kept but only one version can be marked as the primary or good copy. For a more complete description of duplicate page resolution see Resolving Duplicates During Validation of New Records.

The following example of a *Fax/Refax List* shows the outstanding problems for patients 125003 and 125004, which have been marked with a refax request. The QC report format shown here is the default format for a study database defined with field descriptions of 25 characters in length. If 40 character field descriptions are used, the QC report format is slightly different as a result of space constraints.

Figure 10.8. The Fax/Refax List from a DataFax Quality Control report.

FAX/REFAX LIST (Please locate/correct and then fax the following pages of the CRF)			
PATIENT	PLATE	SEQNO	PROBLEM
125003	012	3	missing page
125003	014	3	diastolic blood pressure = " " (missing)
125003	015	3	medication 1: name = "???" (illegible)
125003	015	3	medication 1: dose = "5 sid" (other)
			please clarify dose (mg?) and dosing schedule (sid)?
125004	011	3	date of hospital admission = " " (fax noise)
125004	016	4	overdue visit

Each problem is identified by the patient's study ID number and by the plate and sequence number of the particular CRF page in question. If desired, the plate and sequence numbers can be replaced with a short descriptive label by creating a CRF Page Map as described in *DataFax Study Planning Guide, CRF Page Map*.

Missing CRF pages and overdue assessments are identified by the labels `Missing Page` and `Overdue Visit` respectively. The description `Missing Page` indicates that this page has not been received, although it is known that the patient assessment designated by this sequence number has been performed, because one or more other CRF pages with this sequence number have arrived. The description `Overdue Visit` indicates that the entire patient assessment designated by this sequence number is now overdue. Patient assessment scheduling and an overdue allowance for each visit are specified in the study Visit Map created using **DFsetup**.

Problems which have been flagged with QC notes during data validation are identified by a short description of the data field, the current value of the data field (as entered from the CRF), the type of problem, and optionally by a brief one or two line query from the central data management team.

The short description comes from the description of the data field entered during variable creation in **DFsetup**. The current value appears in quotation marks. For text fields, only the first 13 characters will be printed. The problem type, which appears in brackets at the end of the line, was added when the QC note was created. The standard options include the following:

- **Missing.** The data field was not completed.
- **Illegal.** The recorded value failed the legal values test.
- **Inconsistent.** The value is not consistent with some other data item(s).
- **Illegible.** The data or text recorded in this field could not be read.
- **Fax Noise.** The data field was obscured by poor fax transmission.
- **Other.** This category is used to flag all other problems.

Finally, if a query or comment was added to the QC note, it appears directly below the line which identifies the problem. These comments can be several lines long and wrap appropriately at word breaks. However, generally the automatically included problem type, will be adequate. In those cases where further clarification is needed a 1 or 2 line comment is usually sufficient.

10.6.1.3. Part 3: Question & Answer List

The third part of the standard DataFax QC report is used for QC notes which were not marked with a refax request. The description of each problem on this part of the report follows the same format used on the `Fax/Refax List`, but each problem is followed by a 5 line space. Investigators are asked to answer these questions in the space provided and then to fax the completed report back to the study coordinating center. How these returned QC reports are dealt with to get the corrections they contain into the database, is described later in `When QC Reports are Returned`.

Figure 10.9. The Question & Answer List part of a DataFax Quality Control report.

QUESTION & ANSWER LIST (Print your answer below each question and fax back this page)

PATIENT	PLATE	SEQNO	PROBLEM
125003	016	3	patient to continue = "no, moved to LA" (other) Is patient willing to be followed by a study physician in LA?
125004	014	1	history of myocardial infarction = "yes" (other) How long ago was this MI?

10.6.2. Before Creating New Quality Control Reports

Before creating new QC reports, you should make sure that the study database is as up to date as possible. Otherwise, the reports created may include requests for missing data and clarifications which have already been received.

Suggested checks include the following steps:

1. Have all fax arrivals been entered into the database?

Use **DFstatus** to see if there are any new records awaiting validation or currently being validated by another user. If there are, these records should be validated to at least level 1 before a QC report is created. It is possible that some of the new faxes are the missing pages, overdue visits or refax requests issued on a previous QC report.

2. Are all of the QC notes sensible?

We recommend that one person, typically a clinical research associate (CRA), take responsibility for reviewing all QC notes which have been added by data entry staff at lower validation levels, before they are formatted into a QC report. Some of these notes might prove unnecessary and should be deleted (e.g. because the CRA understood something printed by the investigator which was unclear to the data entry clerk). Others may need to have a comment added or reworded before the query goes into a report. The easiest way to accomplish this is to define a validation level (e.g. 4) for QC editing. Then the CRA can use **DFvalidate** (in validation mode at this specified level) to retrieve all **dirty** records which are below the specified validation level. The QC notes attached to each of these records can then be reviewed and modified or deleted as necessary. Upon completion each data record should be signed off (**clean** or **dirty**) even if no changes were made. This will raise the record's validation level to the level specified for QC editing. As a final check the CRA can use **DFstatus** to make sure that all **dirty** records have been raised to the QC editing validation level, before QC reports are created and sent to the study investigators.

3. Have all replies to previous QC reports been entered into the database?

If you include a `Question & Answer List` in QC reports, you will want to make sure that all replies have been dealt with before a new QC report is created. Investigators may also print notes on other parts of a QC report and fax them back (e.g. to tell you why a page or visit is missing). QC reports are bar coded with Plate number 501 (a DataFax reserved plate number) when they are created. This allows returns to be tracked just like any page of the CRF. We recommend that the same person who reviewed new QC notes (step 2. above) take responsibility for reviewing returned QC reports and making corrections to the database. This might be accomplished by automatically printing these reports when they arrive and having them delivered to the QC monitor. Alternatively the QC monitor could retrieve all copies of Plate 501 which have been validated to level 1 by data entry staff, print them and then sign them off to raise them to the QC editing validation level. Then with the printed QC reports in hand, **DFvalidate** can be used to retrieve the records referenced in the report to make corrections to the data records and to resolve the QC notes. Any pages which are declared permanently missing (e.g. because a test was not performed or a visit was missed) should be recorded in **DFidl**. This will remove them from future QC reports.

4. Is the study centers database up to date?

The study centers database is created when the study is setup, but investigators may come and go or change their fax number, so this database must be kept current. It is located in `DFcenters` in the `study lib` directory and is created and modified by selecting Center DB from the Create menu in **DFsetup**.

5. Is the study Visit Map up to date?

The QC update program uses a Visit Map which is created and modified by selecting Visit Map from the Create menu in **DFsetup**. The Visit Map identifies the expected scheduling of each patient assessment, the overdue allowance permitted before a visit is flagged as overdue on QC reports, and the plates of the CRF which are to be completed and faxed for each patient assessment.

The Generic DataFax report **DF_ICvisitmap** can be used to verify the integrity of a visit map. It is recommended that **DF_ICvisitmap** be executed subsequent to making any change in the study setup.

6. Is the study Page Map up to date?

The creation of a study Page Map is optional. It can be used to replace plate and sequence numbers with a more descriptive label on QC reports. For example, Plate 3 Sequence 1 might be replaced with `Baseline Demographics`. To create or modify a study Page Map select Page Map from the Create menu in **DFsetup**.

7. Do you have exclusive access to the database?

Use **DFstatus** to see whether anyone is currently using any DataFax tools. Although the creation of QC reports does not require exclusive access to the database, it is recommended. Exclusive access is recommended to ensure that all information on the reports will be complete and accurate, and also because the QC reports program rewrites the QC database to include the name of the QC report that each unresolved QC note has been written to.

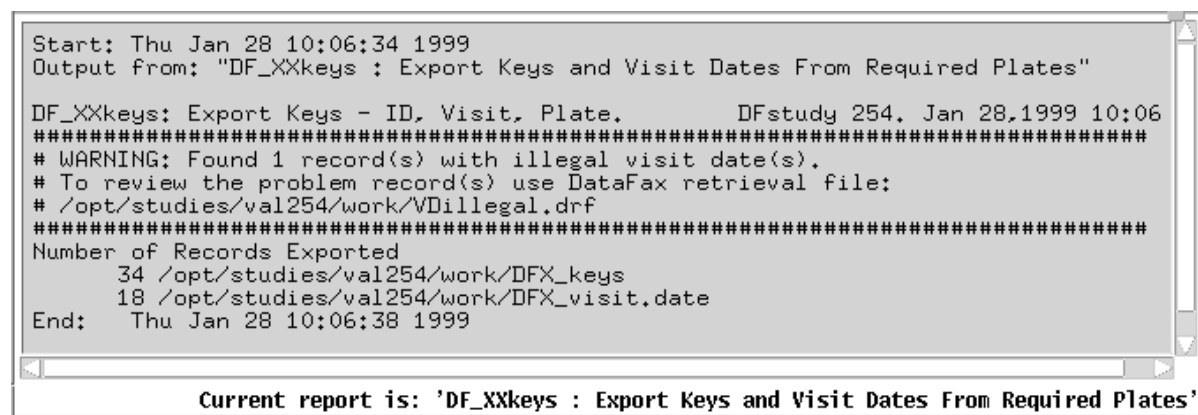
8. Are there any erroneous dates in the database?

If you plan to create a `Patient Status Summary`, you should first attempt to correct any obvious errors in the visit dates recorded in the study database. Otherwise such errors could lead to incorrect listing of a patient's entry and last follow-up dates, and to incorrect calculation of the next scheduled follow-up date.

The DataFax Generic report **DF_XXkeys** extracts all date fields that have been defined with the **VisitDate** style from the list of required plates for each visit, as specified in the study Visit Map. It then checks for both illegal dates (i.e. dates which fail the legal range test specified through **DFsetup**), and for date inconsistencies (i.e. pages of the CRF with the same visit number but different visit dates).

When **DF_XXkeys** is executed, it reports the total number of linked DataFax key fields exported from required plates (as defined in the study Visit Map), and the total number of **VisitDate** fields found. If inconsistencies or illegal visit dates are found, a warning message will be printed as shown in the following example. If you do not execute **DF_XXkeys** before you run **DF_QCupdate**, it will be run automatically by **DF_QCupdate**. However, it is more efficient to run **DF_XXkeys** separately to identify and fix any visit date problems before proceeding to **DF_QCupdate**.

Figure 10.10. A warning message indicates an illegal VisitDate date in the study database.



```

Start: Thu Jan 28 10:06:34 1999
Output from: "DF_XXkeys : Export Keys and Visit Dates From Required Plates"

DF_XXkeys: Export Keys - ID, Visit, Plate.          DFstudy 254. Jan 28,1999 10:06
#####
# WARNING: Found 1 record(s) with illegal visit date(s).
# To review the problem record(s) use DataFax retrieval file:
# /opt/studies/val254/work/VDillegal.drf
#####
Number of Records Exported
      34 /opt/studies/val254/work/DFX_keys
      18 /opt/studies/val254/work/DFX_visit.date
End:   Thu Jan 28 10:06:38 1999

Current report is: 'DF_XXkeys : Export Keys and Visit Dates From Required Plates'

```

Inconsistent Dates. A **VisitDate** inconsistency arises when **DF_XXkeys** finds 2 or more different **VisitDate** dates associated with the same visit sequence number for a particular patient. All inconsistencies are written to a DataFax retrieval file named **VDincon.drf** in the study work directory. This file can be used to retrieve all data records (and CRF pages) with **VisitDate** inconsistencies in **DFvalidate**. You can also display and print a formatted list of these records by running **DF_ICvisitdates**.

Illegal Dates. Illegal **VisitDate** dates are those which have failed the legal values test which was specified when the **VisitDate** style was defined in **DFsetup**. Illegal dates are written to the DataFax retrieval file **VDillegal.drf** in the study work directory. **DF_ICvisitdates** prints a list of any illegal dates after the list of visit date inconsistencies, as shown in the following example.

Figure 10.11. Output from the DataFax Generic report DF_ICvisitdates shows an illegal VisitDate.

```

DF_ICvisitdates: Inconsistent+Illegal VisitDates. DFstudy 254. Jan 28,1999 10:11
NO VISIT DATE INCONSISTENCIES

ILLEGAL VISIT DATES
Validation Retrieval File = /opt/studies/val254/work/VDIllegal.drf
Illegal visit dates detected by DF_XXkeys
DF_XXkeys was run by valid1 99/01/28

PATIENT  SEQNO    DATE    PLATES
99004      21      5

End:   Thu Jan 28 10:11:19 1999
I

Current report is: 'DF_ICvisitdates : List Most Recent QC Export Errors'

```

9. Are there any erroneous key field identifiers in the database?

The DataFax Generic report **DF_ICkeys** generates a listing of patients, visits, and plates that are not referenced by **DFcenters** or **DFvisit_map**. Output from this report aids in identification of data entry errors for patient identifiers. This report should be executed and reported problems resolved before running **DF_QCupdate**. Doing so will ensure that no QC notes are registered in the error monitor report, that no visits are spuriously flagged as overdue, and that no plates are spuriously flagged as missing.

10. Run the DataFax Generic report **DF_QCupdate**

DF_QCupdate should be executed before new QC reports are created. The program reads the entire study database to identify missing plates and overdue visits and adds them to the QC database. It also removes old QC notes identifying missing plates and overdue visits if it finds that they are now in the study database or registered in **DFIdl**. **DF_QCupdate** also checks for conditional plates and will print an error message and terminate if the condition refers to a plate which is not defined or if the condition refers to a field beyond the end of the plate. It also checks for duplicate Missing Page QC notes that may result if a Conditional Plate Map (**DFcplate_map**) has been defined for the study.

DF_QCupdate also updates the DataFax file **DFX_schedule** which is used to create the patient status/scheduling section of QC reports. Changes to the centers database may mean that the information in **DFX_schedule** is no longer up-to-date, resulting in errors or omissions from this section of the QC reports. To ensure that the most accurate and up-to-date information is being used, always remember to run **DF_QCupdate** prior to QC report creation.

Because **DF_QCupdate** must read the entire database and perform many date calculations it takes longer to run than most other DataFax programs; and the time required will increase as the database grows.



Note

Although exclusive access to the database is not required to run **DF_QCupdate**, it is suggested. **DF_QCupdate** rewrites the QC database and requires access to the various study configuration files such as the centers database and visit map. If the information needed by **DF_QCupdate** is not readily available, the QC reports may be created with errors and/or omissions.

If all goes well when running **DF_QCupdate** you should see the following list of messages.

Figure 10.12. A series of messages indicate successful completion of DF_QCupdate.

```

DF_QCupdate: Update Quality Control Database.      DFstudy 254. Jun 08,2005 13:36
1. Checking permissions on study 254, Acceptance Test Study
2. Reading study configuration files
3. Exporting current overdue visit and missing plate QC notes
4. Exporting required data from study database:
   -exporting visit dates and keys from required plates
5. Checking for overdue visits, missing plates and unexpected records
   Missing Pages      Update =    0    Total =    1
   Overdue Visits     Update =    0    Total =    0
   Unexpected Records Update =    0    Total =    1
   No changes required to the QC database
6. Done

```

After the above steps have been completed you will be ready to run **DF_QCreports** to create new Quality Control reports which accurately reflect the current status of the study database.

10.6.3. Types of Quality Control Reports

DF_QCreports creates 2 basic types of QC reports:

1. **External.** An external QC report is created for transmission back to each of the clinical sites. These reports summarize the status of all patients being followed at the site and include all of the QC notes which identify problems that need to be resolved.
2. **Internal.** Internal QC reports, on the other hand, are strictly for internal use and are not sent to the clinical sites. Unlike the external reports which only include those patients being followed at a particular clinical site, internal QC reports can include all patients or a group of special interest.

10.6.3.1. External Quality Control Reports

Each page of an external QC report is identified by the center's ID number, the report creation date and the report page number, in the following format: `ccc-yyymmdd-pp`. Thus for example, 088-950822-01 would appear on page 01 (01) of a Quality Control report created for Center # 88 (088) on August 22, 1995 (950822).

In trials of long duration a new Quality Control report is typically created and faxed to each clinical center on a fixed schedule (e.g. once a week). In trials of shorter duration quality control reports may be sent more frequently; perhaps at the end of each day that new information arrives from the center. The scheduling of report creation and faxing is up to each study coordinator but because reports are named by date you should not send more than one QC report per day, otherwise you will have 2 different QC reports with the same name.



Note

If you generate more than one Quality Control report per center per day, DataFax will issue a warning message and will not generate another report for that day.

You must have exclusive access to the database when you create external reports because the QC database is re-written to update the status and report number for all QC notes which are included in the newly created reports.

Each QC note has a data field which holds its current status, which can be any one of the following 6 options:

- **new, not in QC report.** Indicates that the note has not yet been written to an external QC report.
- **in QC report, not sent.** Indicates that the note has been written to a QC report but that this report has not yet been faxed to the clinical center. This is the status that **DF_QCreports** will set for all QC notes which it writes to newly created external reports.
- **in QC report, sent.** Indicates that the note is in an external QC report which has been faxed to the clinical center, but no correction has yet been received.
- **resolved, corrected.** Indicates that the CRF has been corrected and the QC notes have been resolved.
- **resolved, not available.** Indicates that the investigator has indicated that the requested information will not be available, and the QC note has been resolved.
- **resolved, not relevant.** Indicates that no corrections were made to the CRF because the QC request was not really relevant in the first place, and the QC note has been resolved.

Thus, the status of each QC note changes as it progresses through the quality control cycle. The current status of all QC notes can be checked using **DFqc**.

In addition to current status, each QC note has a data field which holds the patients center ID and another field which holds the date of the most recent QC report. Together these 2 fields make up the QC report ID number written at the top of each page of a QC report. When old unanswered QC notes are written to a new QC report the date field gets overwritten with the new QC report date, and the status field is set back to status **in QC report, not sent**. The center ID field in QC notes gets overwritten if the patient moves to a new center.



Note

The current status and QC report date fields are reset on all QC notes each time that **DF_QCreports** is executed. Since **DFvalidate** provides easy access to all records associated with a specified QC report ID number, and since **DFqc** can be used to review the current status of all QC notes, you will not want to create external QC reports unless you really intend to send them out to the clinical centers and thus want to have the unanswered QC notes moved to this new report and have the current status field re-set to **in QC report, not sent**.



Note

External QC reports only include those QC notes that have been marked with usage type external. If an external QC note has been included in an external QC report, and the note's usage type is changed from external to internal in **DFvalidate**, the Reply Type, QC Status, and QC Report fields will not be removed on the external QC report. If you want to change the usage type of a QC note from external to internal, first delete the external QC note in **DFvalidate**, before adding an internal QC note to the data field in question.

10.6.3.2. Internal Quality Control Reports

Unlike external QC reports, internal QC reports can be created at any time because they do not alter the QC database in any way. Also, because the QC database is not re-written it is not necessary to lock the database to create an internal QC report. However, if other users are working in **DFvalidate** the study database will be undergoing additions and/or modifications and thus your report may not be entirely accurate. All internal QC reports are written to the study directory `reports/QC/internal`.

10.6.4. Quality Control Report Creation Options

There are many options which can be used in combination to produce a wide variety of external and internal QC reports.

As for all programs managed under **DFreports**, options to **DF_QCreports** are selected from the Options button, or by typing them directly into the Options list.

Figure 10.13. Some possible options used in running DF_QCreports.

A screenshot of a software interface. On the left, there is a button labeled 'Options:' with a small square icon to its right. To the right of this button is a rectangular text input field. Inside this field, the text '-o 21 -c 10~19' is entered.

The first argument, `-o 21`, specifies that QC notes which have been sent to the clinical sites in previous QC reports are not to be sent again unless 21 or more days have passed since the previous QC report. This overdue allowance can be used to give investigators a chance to reply to previous requests before you ask for them again. The second argument, `-c 10~19`, states that reports are to be created for Centers 10 through 19. Any unused centers numbers in this range (i.e. any values not defined in the centers database) will be ignored. The following items should be considered when generating external Quality Control reports.

- Which clinical centers are to be included?
- Which patients are to be included?
- Which parts of the standard QC report are to be included?
- How long should we wait for a response before asking for clarification again?
- Should an internal QC report be created?

Each can be specified using the available **DF_QCreports** options.

`-i srqQPc`

Select the parts to be included in each QC report.

`s`: Patient Status Summary

`r`: Fax/Refax List

`q`: Question & Answer List

`Q`: Use a new page for each patient

`P`: Refax and Q&A notes appear together on a new page for each patient

`c`: Cover sheet and messages

The default is `-i csrq` which includes a cover sheet (if `QCcovers` and `QCmessages` exist), Patient Status Summary, Fax/Refax List, and Question & Answer List.

The `P` and `Q` flags alter the Fax/Refax List and Question & Answer List sections of the report. The `P` flag puts all refax and Q&A notes on a separate page(s) for each patient. The `Q` flag creates a new page for each patient for Q&A notes only. Thus for example `-i srQ` will create the standard Patient Status Summary and Fax/Refax List followed by a separate page(s) for each patient who has unresolved Q&A notes, and `-i sP`, will produce reports containing the patient status list for all patients followed by a separate page(s) of QC notes for each patient who has outstanding unresolved QC notes. Some combinations, e.g. `-i rqP`, represent an inconsistent request and are thus illegal.

Cover sheets and messages can be generated without a QC report using `-i c`. This can be used to create a broadcast notice to be sent to all clinical sites.

`-b v/p`

Page breaks. This options supplements `-i PQ` by placing a page break between visits within patients (`-b v`), or by placing a page break between plates within visits within patients (`-b p`).

`-c #, #-#`

Include only the specified center numbers.

`-C #, #-#`

Include only the specified patient IDs. Only create reports for study centers that include the specified patients, and include only the specified patients in those reports.

`-cd yy/mm/dd-yy/mm/dd, -cd
yyyy/mm/dd-yyyy/mm/dd`

Select only QC notes created within the specified date range, inclusively.

`-f name`

Create a single internal QC report (not for transmission to study sites) and save it to the named file in the `$REPORTS_DIR/QC/internal` directory. All selected QC notes are included in the same internal QC report. If a QC note includes a reply to the query, the reply text will be printed in the report directly below the query text.

`-q QCtypes,tag`

This option, which works with internal reports only, is used to select QC notes by type. Any combination of the following options may be specified:

I: all internal QCs

E: all external QCs

R: all resolved QCs

U: all unresolved QCs

P: all pending QCs

iu: internal unresolved QCs

ir: internal resolved QCs

ip: internal pending QCs

eu: external unresolved QCs

er: external resolved QCs

ep: external pending QCs

all: all internal and external QCs of all statuses
(equivalent to `-q er,ep,eu,ir,ip,iu`)

The inclusion list is comma delimited with no spaces. The default is `eu` (i.e. external unresolved QCs).

The keyword `tag` may also be used with the `-q` option to show the status and usage of each QC note in the report. If `tag` is specified, each QC will be tagged with its status and usage following the problem type, using the same strings specified with the `-q` option. For example, an external unresolved missing value QC will be tagged as `(Missing Value-eu)` in the internal QC report.

`-o #`

Include old QC notes only if they were previously appeared in an external QC report that was created more than specified number of days ago. The default is 0 which includes all unresolved QC notes in every QC report.

`-p 0/1/2/3`

Limit the Patient Status Summary to specified patients from the following types:

0, all patients (the default)

1, patients with ongoing follow-up

2, patients with unresolved QCs

3, patients with ongoing follow-up, unresolved QC notes, or both

`-e b/f/b1/f1/x`

Set the "Entry Visit" displayed on the Patient Status Summary where

`b` requests the baseline from the last cycle received

`f` requests the first scheduled visit (P or B) from the last cycle received

`b1` requests the baseline from cycle 1

`f1` requests the first scheduled visit (P or B) from cycle 1

`x` requests the first screening visit

`-l lastv/lastsv`

Set the "Last Visit" displayed on the Patient Status Summary where

`lastv` requests the last visit received with a Visit-Date (any visit type), or

`lastsv` requests the last scheduled visit received (types: P,B,S,T,F)

`-s no/yes`

Include patients on the Patient Status Summary who only have screening visits (no or yes)?

The default for this option is `yes`, meaning that all patients will be included. If `no` is selected those patients who have only completed screening visits will not be included, unless the screening cycle has been terminated and a visit in an in-study cycle is now required.

`-P #, #-#`

Include only QC notes on the specified plates

`-S #, #-#`

Include only QC notes on specified visits

`-T #, #-#`

Include only QC notes with problem codes from the list: 1=missing, 2=illegal, 3=inconsistent, 4=illegible, 5=fax noise, 6=other, 21=missing page, 22=overdue visit, 23=edit check missing page

`-v #, #-#`

Include only QC notes at the specified validation levels. This does not apply to missing plate or overdue visit QC notes. These notes are output regardless of their validation level.

`-F visitlist:platelist:fieldlist`

Select QC notes that exist on the specified data fields. A single value, range and/or range list may be specified for any of *visitlist*, *platelist*, or *fieldlist*. A blank *visitlist*, *platelist*, or *fieldlist* indicates that all values satisfy the criteria. Multiple specifications are allowed with `-F`, each separated by a single space or a '+'. Spaces are not allowed within a specification.

`-N "string1" "string2" ...`

Select QC notes containing the specified string(s) in their *Note* field. A single string or multiple strings may be specified. Multiple strings are each separated by a single space and QC notes are included in the QC report if a match is found on one or more of the specified strings. String-matching is case-sensitive. The double quotations around the string specification are optional.

`-idfmt #`

Leading zero-pad patient ID numbers to the requested number of digits. The default is no zero padding of patient IDs.

`-dtfmt fmt`

Set the output format for dates displayed in the Patient Status Summary.

The date format can be set as illustrated in the following examples for Jan 15, 2000:

`-dtfmt "mm/dd/yy" displays 01/15/00`

`-dtfmt "dd/mm/yyyy" displays 15/01/2000`

`-dtfmt "yyyy/mmdd" displays 2000/0115`

`-dtfmt "mmm dd,yyyy" displays Jan 15,2000`

`-dtfmt "yyyymmmdd" displays 2000Jan15`

Any combination of *dd*, *mm* or *mmm*, *yy* or *yyyy* and delimiters can be used. The total space allotted for the visit label and date is 23 characters, including the 2 characters for ":" between the label and date. **DF_QCreports** calculates the space available for the visit labels based on the specified the date format and will truncate any visit label that is too long.

<code>-probfmt 1/2</code>	Number queries within each QC report and put the problem description first, rather than at the end of the line, where <ol style="list-style-type: none"> 1 the default format is preserved (queries are not numbered and the problem description follows the patient ID, plate and sequence number). 2 queries are numbered and problem description appears first in the description of each QC note.
<code>-rline</code>	Draw a line between patients in the refax list; otherwise, patients are separated by a blank line.
<code>-rspace</code>	Leave a blank line (for comments) between each query in the refax list. Without this option, the default appearance is that no blank lines separate consecutive queries.

10.6.4.1. Changing Quality Control Report Headings

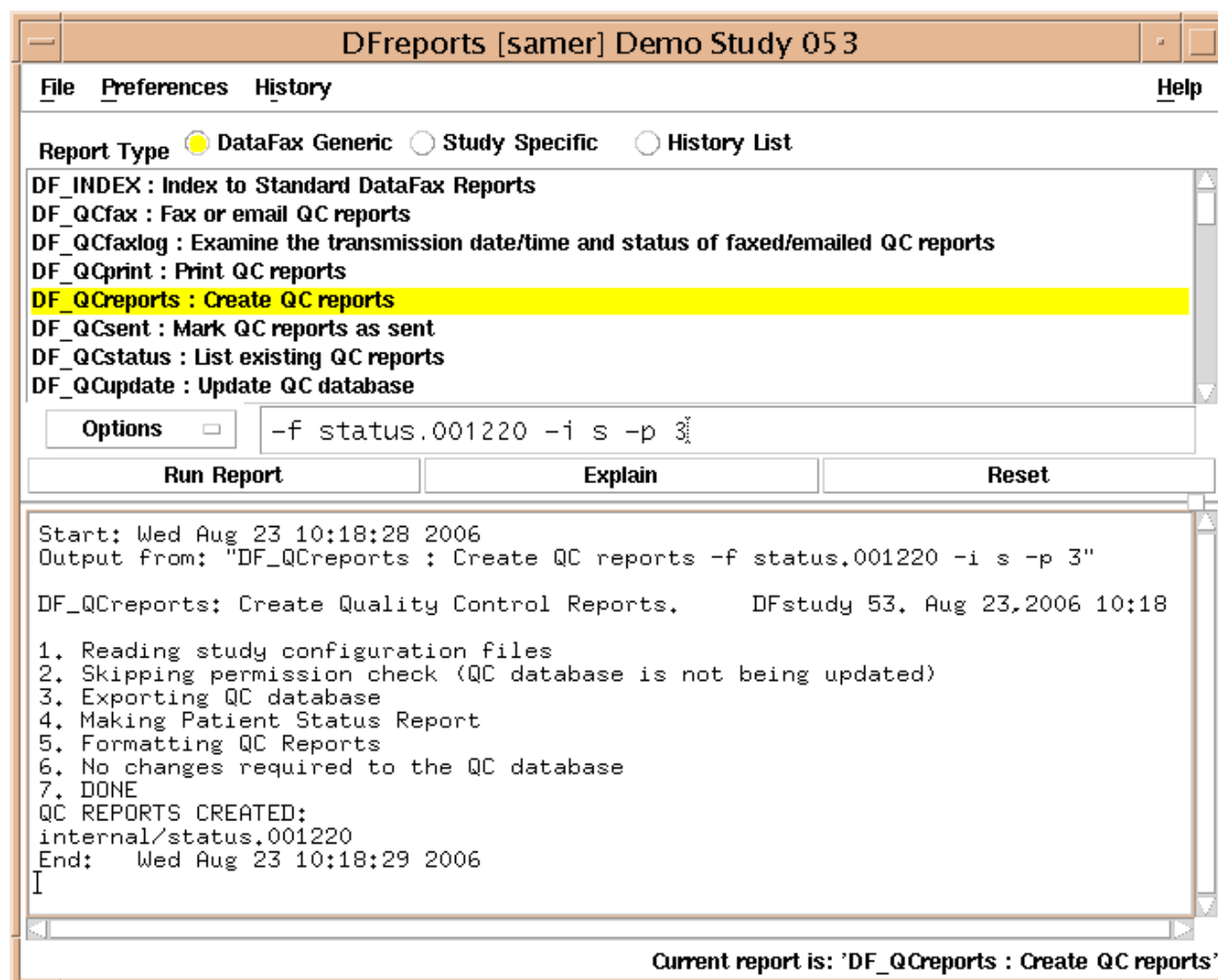
Quality Control Report headings may be changed by either creating a file called `QCtitles`, or by editing the file `DFqcps.prolog`.

1. **QCtitles.** `QCtitles` is a file that can be created in your study's `lib` directory to change the title and subtitles of external or internal QC reports. It is fully described in *DataFax Programmer Guide, QCtitles - QC report titles*.
2. **DFqcps.prolog.** The file `DFqcps.prolog` resides within the `lib` directory of a DataFax installation. This file contains the formatting information used for quality control reports for all DataFax studies. The necessary headers can be located and modified using a text editor. Changing this file in this directory makes the change occur for all DataFax studies. To make the change for a single study, copy `DFqcps.prolog` from your DataFax installation directory to the appropriate study `lib` directory and edit that file instead.

10.6.5. Executing Quality Control Reports

After selecting **DF_QCreports** from the scrolling list of report names and specifying any desired options, click Run Report to execute the program. As **DF_QCreports** runs it prints status messages to the output window.

Figure 10.14. The status message written for the creation of an internal Quality Control report.



In step 3., **DF_XXkeys** is executed. This step is skipped if the database has not changed since the last execution of **DF_XXkeys**. When external QC reports are created the database is locked (step 3.) and the QC database is rebuilt (step 6.) to update the current status of each QC note and to identify the current QC report to which each QC note has been written. As shown in the above example these steps are not required when creating internal QC reports.

There are 3 ways to customize the appearance of external QC reports:

1. Modify QC report titles.
2. Add cover pages.
3. Include messages on cover pages.

All three can be implemented by creating a simple text file for each in the study lib directory. A detailed description can be found in *DataFax Programmer Guide*, *DataFax Study Files*.

10.6.6. Checking Quality Control Reports

When **DF_QCreports** terminates it displays a list of the reports it created. This list is saved in file **QC_NEW** under the study **reports/QC** directory. You can examine this file at any time, to see a list of the most recently created reports, by running **DF_QCstatus**.

External QC reports are named by center and creation date using format **ccc-yyymmdd**. Thus, for example file **025-950708** would contain the QC report created for Center 25 on July 8, 1995. All external QC reports are written to the study **reports/QC** directory when first created and are moved to the **reports/QC/sent** directory when they are registered as having been successfully faxed to the clinical sites.

Internal quality control reports can have any name you care to specify, and are written to the study **reports/QC/internal** directory.

After creating QC reports you can examine them either by viewing the raw ASCII QC report files or by running **DF_QCprint** to get a paper copy formatted exactly as the report will appear when faxed to the clinical centers. These 2 methods are described below.

1. **Viewing QC Reports on Screen.** Program **DF_QCview** is used to view QC reports in the **DFreports** window. With no options the program will display all reports listed in **QC_NEW**. Thus after creating a new set of reports, you can view them by selecting **DF_QCview** from the scrolling list of DataFax Generic reports, and clicking Run Report, without specifying any options.

Options for **DF_QCview** include the following:

Figure 10.15. A list of options available for the DataFax Generic report DF_QCview.

Options:
<u>C</u> lear Options
-u <report number> ...display an unsent report
-s <report number> ...display a sent report
-i <filename> ...display an internal report

2. **Printing QC Reports.** Program **DF_QCprint** is used to print QC reports to the default DataFax printer. With no options this program will print all reports listed in **QC_NEW**. Thus after creating a new set of reports, you can print them by selecting **DF_QCprint** from the scrolling list of DataFax Generic reports, and clicking Run Report, without specifying any options.

Options for **DF_QCprint** include the following:

Figure 10.16. A list of options available for the DataFax Generic report DF_QCprint.

Options:
<u>C</u> lear Options
-u <report number(s)> ...print unsent reports
-s <report number(s)> ...print sent reports
-i <report name> ...print internal report

Only one of these 3 options can be specified at a time, and the full report name must be specified for each report that you want to print.

Figure 10.17. The option list needed for DF_QCprint to print 2 external QC reports which have not yet been registered as successfully faxed to their respective clinical sites.

Options: <input type="checkbox"/>	-u 001-950215 122-950215
-----------------------------------	--------------------------

10.6.7. Faxing Quality Control Reports

QC reports can be sent to the clinical centers automatically by DataFax, or manually by study management staff. To send reports manually use **DF_QCprint** to produce a paper copy of each report and send it by fax, mail or courier to the clinical centers. When the manual approach is used each successfully transmitted QC report must then be registered as sent using **DF_QCsent**, so that DataFax will know that the QC notes have been transmitted to the clinical centers.

This entire process can be automated using **DF_QCfax**. This program will send the QC reports to the clinical sites through the same fax modem(s) used to receive CRFs, and will also track them and automatically execute **DF_QCsent** to update the QC database if they are transmitted successfully. If they are not transmitted successfully, **DF_QCfax** may be executed in **-b** mode to automatically re-send those reports that have failed.

If **DF_QCfax** is executed with no arguments it will immediately attempt to fax all newly created QC reports (listed in **QC_NEW**) to the appropriate clinical sites, using the fax numbers or email addresses listed in **DFcenters**. You can send each report to more than one destination fax machine or email address by including multiple fax numbers and/or email addresses in **DFcenters**. In **DFcenters** multiple numbers and/or email addresses are entered into the fax number field separated by a space. When multiple fax numbers/email addresses are present, DataFax only considers the first one when deciding whether or not to execute **DF_QCsent**. Thus the first entry should be that of the individual who is expected to respond to the report while subsequent numbers might be for others who are to receive copies of transmitted QC reports.

Transmission can be delayed by using the **-w** option. For example: **-w 02:00** will delay transmissions until 2 AM, **-w 15:30** will delay transmission until 3:30 PM, and **-w midnight** will delay transmissions until midnight. If you have many QC reports to transmit or you are trying to reduce telephone costs, transmissions can be delayed to avoid tying up the modem when investigators might be trying to submit CRFs and also to take advantage of lower long distance telephone rates.

You can send reports to a specified fax number or email address instead of the number found in the centers database by including **-f** followed by the desired fax number or email address in the options list. Remember to include the full fax number just as you would dial it from a fax machine. Dashes may be included, but are not necessary, e.g. **-f 1-905-574-2837**. An email address must be specified using the notation **mailto:email_address**, where the prefix **mailto:** is fixed and required and **email_address** must be a valid email address.

If specific reports are not named in the argument list DataFax sends all reports listed in file **QC_NEW**, but only if they have not already been sent and moved to the **QC/sent** directory. If the **-a** argument is used all unsent QC reports which reside in the **QC** directory will be faxed. If **DF_QCfax** has been executed and some QC reports have failed to be transmitted, the **-b** option can be used with **DF_QCfax** to re-send only those reports that have failed. To refax another copy of a report which has already been sent (and moved to the **QC/sent** directory) use the **-s** option followed by the report name. And to fax an internal QC report use the **-i** option followed by the name of the internal QC report (which must be located in the **QC/internal** directory).



Use of -a

Use the `-a` option carefully. Execution of **DF_QCreports** that is not followed by **DF_QCfax** will leave unsent reports in the QC directory. Specifying `-a` with **DF_QCfax** will subsequently send **all** QC reports found in the QC directory.

Finally, the `-t` option can be included in the argument list to signal that you only want to test **DF_QCfax** and not actually send anything. With this option **DF_QCfax** will display what it would have done in the **DFreports** window. This display includes the name of the report and destination fax number for each report named in the transmission list. Note that currently it does not include any indication of when it would have tried to send the report, and thus you can not test the `-w` option.

See *DataFax Standard Reports Guide*, **DF_QCfax** for further options and examples of **DF_QCfax**.

10.6.8. Checking Fax Transmission Status

It is important to check to see whether the QC reports submitted to **DF_QCfax** actually made it to the intended destination. By default **DF_QCfax** is set to retry unsuccessful transmissions twice. A total of 3 attempts will be made over a 30 minute interval (10 minutes between retries). The result, whether success or failure, is written to a log file named **SEND FAX.log** in the study **reports/QC** directory [9].

The **SEND FAX.log** can be examined using **DF_QCfaxlog**. When executed with no arguments **DF_QCfaxlog** reports on the status of all QC reports listed in **QC_NEW** (i.e., the list of most recently created QC reports). The `-l` option followed by an integer displays the specified number of lines from the end of **DF_QCfaxlog**, and the `-m` option followed by a string lists the status of all lines in **DF_QCfaxlog** which match the specified string. For example, `-m 001-` will list the status of all QC reports submitted to **DF_QCfax** for transmission to Center 001, and `-m -921215` will list the status of all QC reports created on Dec 15, 1992 and submitted to **DF_QCfax**. *DataFax Standard Reports Guide*, **DF_QCfax** contains additional examples.

10.6.9. Registering Successfully Faxed QC Reports

When a QC report is transmitted manually to the clinical sites (by ordinary fax or surface mail) they must be registered with DataFax using **DF_QCsent**. If you use **DF_QCfax** it will track the outgoing QC reports and run **DF_QCsent** automatically as soon as it is notified that the fax was successful. But if you print and fax the QC reports manually you must run **DF_QCsent** yourself.

DF_QCsent rewrites the QC database marking all QC notes in the specified QC reports as **sent**, and then moves the reports to the **QC/sent** directory. Upon completion it prints a summary of all changes that have been made. This is necessary to keep the status of QC notes current. Otherwise the QC status report which is created by **DFqc** will not be accurate. Also the `-o` option on **DF_QCreports**, which allows you to specify an overdue allowance before repeating questions, will only skip QC notes which have been registered as **sent** within the specified allowance.

To register all of the QC reports listed in **QC_NEW** (i.e. all those reports most recently created), select **DF_QCsent** from the scrolling list of DataFax Generic reports and click Run Report, without specifying any options.

10.6.9.1. DF_QCsent Options

To register one or more specified QC reports as **sent**, use the `-s` option followed by the full name of each of the reports to be registered, as shown in the following example.

Figure 10.18. The `-s` option is used to register QC reports as `sent`.

Options: <input type="checkbox"/>	<code>-s 001-950215 033-950215 145-950215</code>
-----------------------------------	--

Alternatively, if the reports to be registered as `sent` are listed in an ASCII file (with one QC report name per line), this file can be submitted to **DF_QCsent** using the `-f` option.

Thus for example if you wanted to register all of the reports listed in `QC_NEW` as `sent` except for two (which for some reason could not be sent to the clinical sites) you could copy `QC_NEW` to another filename (e.g. `faxed.950215`), edit this file to remove the 2 failed reports and then use the `-f` option to submit the list of successful reports to **DF_QCsent**, as shown in the following example.

Figure 10.19. The `-f` option is used to submit successfully faxed QC reports to **DF_QCsent.**

Options: <input type="checkbox"/>	<code>-f faxed.950215</code>
-----------------------------------	------------------------------

If you make a mistake and need to reset QC reports back to `unsent`, use the `-u` option followed by a list of the reports to be de-registered as shown in the following example.

Figure 10.20. The `-u` option registers QC reports as `unsent`.

Options: <input type="checkbox"/>	<code>-u 001-950215 033-950215</code>
-----------------------------------	---------------------------------------

10.6.9.2. When QC Reports are Returned

If QC notes are flagged with a refax request when they are created, they will be resolved when the CRF page is corrected and refaxed (as described in *DataFax User Guide, Resolving Duplicates During Validation of New Records*). However, the clinical site might write answers to your questions directly on the QC report and fax it back to you. This is expected when QC notes are assigned to the `Question & Answer List`, but investigators might also write in the margins of the `Fax/Refax List` and send that back as well. Your job is to read their response, enter it into the data base, and resolve the QC notes. This can be accomplished as discussed in the following descriptions.

10.6.9.2.1. Validating Returned Quality Control Reports

DF_QCprint and **DF_QCfax** print the report name (`ccc-yyymmdd`) and bar codes on each page of all external QC reports. Bar coding includes: the DataFax study number, plate number 501 (which is reserved for QC reports), and the number of each page within the report. Returned pages of a QC report will thus arrive in **DFvalidate** as new records just like any other page of the CRF. When they do simply register them by entering their external report name (`ccc-yyymmdd`) and sign them off to have them written to the study database. The external report name becomes the ID number of that report for retrieval purposes within **DFvalidate** and **DFviewer**.

10.6.9.2.2. Resolving QC Notes on Returned Quality Control Reports

Usually the CRA or data manager who had the final say in issuing the QC notes will take responsibility for reviewing returned QC reports, making corrections to the database, and resolving the QC notes.

A suggested approach (SOP) is as follows:

1. Select a validation level to which returned QC reports (plate 501) will be raised when they have been reviewed and any necessary corrections made to the study database.

This should be a validation level higher than validation level 1.

2. Check for newly returned QC reports.

Periodically, check to see if there are any new returned QC reports which have been validated by the data entry staff, i.e. which are below the QC report review validation level. This can most easily be done using **DFstatus**.

3. Retrieve newly returned QC reports.

Using **DFvalidate** (in `Validate` mode at the QC report review validation level) retrieve all returned QC reports (plate number 501) which are below the QC report review validation level. Print all of the pages retrieved and then use `Validate Set` to raise them all to the QC report review level.

4. Incorporate the responses into the database.

Now, with the paper copy of returned QC reports in hand, use **DFvalidate** to enter corrections into the study database and to resolve the corresponding QC notes. Also, use **DFidl** to register any assessments which investigators declare as missing and irretrievable for whatever reason.

Correct the database and resolve QC notes . Making corrections to the database using **DFvalidate** is facilitated by using the `By QC Report` retrieval option when building your retrieval set. This option will assemble all CRF pages (and data records) referenced in the specified QC report. You can then proceed to make the corrections and resolve the corresponding QC notes. Don't forget to resolve the QC notes or they will reappear on the next QC report. Also, don't forget to sign off the data records you modify (by setting status to `clean` or `dirty`). The changes you make to data records and QC notes are only written to the database when the data record is signed off. If you move to another data record without signing off the current record, any changes you made to the current record and its QC notes will be discarded.



Important

When a patient is moved from one center to another, DataFax will immediately update the QC notes so that they will belong to the new center number. However, if these QCs have been sent to the original center in a QC report, remember that they will appear in the QC report that was sent to the original center. But, if you attempt to retrieve the the patient's QC notes using the original QC report name (old center number plus report date), they will not be found. This is because QC notes do not store the report IDs in a single field, instead they contain 2 fields, one for the center ID and another for the date a report was last sent. So when the center ID is changed, you can trick DataFax into retrieving the records for a patient who has moved to a new center by specify the new center number plus the last report date when performing a `By QC Report` retrieval.

Register lost CRF pages. . The second type of correction, registering CRF pages as `lost`, is performed whenever a clinical site informs you that a `Missing Page` or `Overdue Visit`, listed on a QC report, will never be available (e.g. the patient missed a visit, refused a test, etc.). This fact is registered in the database by selecting `Lost Data Log` from the DataFax toolbox. If this is not done, subsequent QC reports will continue to complain about the missing page or overdue visit.

The above procedure should be performed on a regular basis, and ideally should be completed before each new set of QC reports is issued. Otherwise, subsequent QC reports may repeat queries which the clinical sites have already responded to. This would certainly annoy them and potentially might cause them to ignore subsequent QC reports all together. If you are having trouble keeping up with the volume, you might want to increase the allowance made for replies to be received before QC notes are reissued in another QC report (see Quality Control Report Creation Options). However, this will not avoid the problem entirely, as a site may have responded late (i.e. past the overdue allowance). Thus every effort should be made to keep up to date and resolve QC notes on all returned QC reports before issuing new QC reports.

10.7. Installing Your Own Reports

In most clinical trials, it is common to create routine study status reports and data integrity checks which are executed at regular intervals throughout the trial. Such routine reports can be created by putting all of the computational steps (data retrieval, analysis and report creation) into a UNIX shell script. This script can then be documented and installed in the study `reports` directory. Once this is done, and tested, it can then be executed by anyone with access to the study's **DFreports** Tool. Of course some analyses must be performed by a skilled statistician each time they are performed, but most of the summary reports created for trial and database management purposes are simple routine summary reports (i.e. tables and graphs). In such cases, setting up a UNIX shell script which performs all of the steps needed to create the report, can relieve programmers and data analysts from those tasks which are simple, repetitive and time consuming and allow them to concentrate on those analyses which require more attention and skill, and which as a result are usually much more interesting.

DataFax Programmer Guide, Writing Your Own Reports includes a description of the location and format of study data files, the quality control data base and study configuration files, as well as the programs available for exporting data from a study database (**DFexport.rpc**, **DFget**, **DFsas**). With these tools programmers can extract the data they need from the study database. They can then write report programs in whatever language they are familiar with (e.g. SAS, C, Perl, UNIX shell scripts, etc.). DataFax does not include yet another report language to learn. You might also want to examine the standard DataFax Generic reports, which are installed in the DataFax `reports` directory for examples of how to construct your own study report programs.

When writing database integrity check programs we recommend that they be designed to create both a formatted report, which can be printed, and a DataFax retrieval file, containing the patient ID, plate number and sequence number of each record that failed the integrity checks. The printed report and DataFax retrieval file can then be used to review the problems, make corrections where possible and flag data fields with QC notes where necessary. It is also possible to have an integrity check program automatically add QC notes to the QC database. However, this is trickier, and runs the risk that QC notes may be added where instead a data entry error should be corrected.

The next page lists a simple example of a UNIX shell script which reports patient entry by center for the past 9 months, as well as the total to date, for one of our current studies.

Only the basic outline of this program is documented with in-line comments, but this should be enough to give you an idea of the level of effort required to produce simple study specific report programs. For those unfamiliar with UNIX shell programming we hope it will be enough to whet your appetite for an exploration of the many useful tools built into the UNIX operating system. You can accomplish a great deal without invoking big systems like SAS, and its really not that hard, honest!

```

#!/bin/sh
# entry9 - dwt created 940915
# RANDOMIZATIONS BY CENTER FOR PAST 9 MONTHS AND TOTAL TO DATE
yy='date +%y'      # yy = current year
mm='date +%m'      # mm = current month
hx=9               # hx = number of months to be printed
# Export all primary randomization records received to date (plate 1)
# $1 is study number
DFexport.rpc -s primary $1 1 - | \
# A simple awk program reads the exported data file and produces the report
nawk -F\| 'BEGIN{YY="$yy"+0; m=MM="$mm"+0; HX="$hx"
# determine which months are to be printed
if(m>=HX) for(i=HX;i>=1;i--) {k[i]=YY*100+m;--m}
else {
    jn = HX - MM    # number of months from past yr
    jm = 13 - jn    # first month from past year
    jy = YY - 1     # past year
    for(i=1 ;i<=jn;i++) {k[i]=jy*100+jm; ++jm}
    for(jm=1;i<=HX;i++) {k[i]=YY*100+jm; ++jm}
}
}
{ # Read each exported record and count cases by month and center
mon=substr($9,1,2); yr=substr($9,7,2)
center=int($7/1000); yymm= yr*100 + mon
n[yymm,center] +=1
mtot[yymm] +=1
ctot[center] +=1
}
END{ # Print Report
printf"RECENT AND TOTAL RANDOMIZATIONS BY CENTER\n"
printf"CENTER "
for(i=1;i<=HX;i++) printf"%6d",k[i]
printf" TOTAL\n\n"
for(i=1;i<=200;i++) { # center number range is 1~200
if( ctot[i]>0 ) {
printf"%5d: ",i
for(j=1;j<=HX;j++) printf"%6d",n[k[j],i]
printf" :%6d\n", ctot[i]
SUM += ctot[i]
}
}
printf"\nTOTAL: "
for(j=1;j<=HX;j++) printf"%6d",mtot[k[j]]
printf" : %5d\n", SUM
}' | sed 's/ 0/ /g'

```

Installing and executing this program from **DFreports** produces the output shown in Example 10.2.

Example 10.2. Sample output from entry9 report

DFreports [ACE]

File Preferences History List Help

Report Type: ☐ Study Specific

entry9: Patient Entry by Center in Past 9 Months and Total
 icheck0: Check Patient Eligibility at Randomization
 icheck1: Check Patient Eligibility at Initial Assessment
 icheckVD: Check Visit Dates for Consistency within Patients
 icheckCR: Check Patient Compliance

Options: ☐

Run Report Explain Reset

Start: Tue Feb 28 07:16:19 1995
 Output from: "entry9: Patient Entry by Center in Past 9 Months and Total"

RECENT AND TOTAL RANDOMIZATIONS BY CENTER

CENTER	9406	9407	9408	9409	9410	9411	9412	9501	9502	TOTAL
3:					1	1	3		2	7
5:				1	1	1		1	2	6
9:							5		1	6
10:			1	1	5	3	2	2	1	15
11:						1		1		2
14:							1		1	2
20:						2		2	1	5
22:		2				2	3			7
23:				3	4	2	1	3	5	18
26:			2	2	6	1	1	1	1	14

Current report is: 'entry9: Patient Entry by Center in Past 9 Months and Total'

[9] There is no reliable method for determining if a QC report sent via email is actually received. Hence, if a QC report is to be delivered by email to a recipient, the system always assumes that the email was also received.

Chapter 11. Data Export Tool

11.1. Introduction

DFexport can be used to retrieve all or selected data fields from any plate in a study database. Retrievals can be viewed on screen in a spreadsheet output window and/or written to a file. **DFexport** includes all of the record selection options available in the DataFax command line program **DFexport.rpc** plus the query by variables feature found in **DFvalidate**. It has the ability to save retrieval criteria as named tasks and provides sort and search capability within the data records output to the spreadsheet window. **DFexport** can only be used to retrieve and view patient data. It does not allow the database to be modified in any way.

As the selection criteria available in **DFexport** correspond to the command line arguments available in **DFexport.rpc**, the documentation for **DFexport.rpc** should be consulted for additional details. It can be viewed by selecting Contents from the Help menu at the top of the main **DFexport** window.

11.2. A Quick Start Guide

11.2.1. Starting *DFexport*

The **DFexport** tool can be started in one of two ways:

1. By clicking the **DFexport** icon in the study toolbox.
2. From the command-line as

```
% DFexport -s study #
```

11.2.2. User permissions

Permissions to use **DFexport** must be granted by an administrator. If a user does not have the necessary permissions, or specifies an invalid study number, **DFexport** will not start and a warning dialog will be displayed.

11.2.3. User interface

At startup, **DFexport** appears as follows:

Figure 11.1. The main window of DFexport

DFexport [valid1] 254 – Acceptance Test Study

File Edit Search Sort Filter Misc Help

Tasks

Name

Selections

Plates

000 – New Data Records
 001 – Blood Pressure Screening Visits
 002 – Patient Entry Form
 003 – Medical History
 004 – Current Medications
 005 – Monthly Follow-up Form
 006 – Patient Evaluation Report

Date Format

☒ Default ☐ Calendar ☐ Julian

Options

☐ Decode Coded Field ☐ Quiet Mode

Lost Code

☒ Use Default Code [*]
☐ Use

Headers

Fields

☐ Key [id, plate, visit, status, validation level]
 Field Number ...

Data Filters

Status ...
 Validation
 Center
 Patient
 Visit
 Creation
 Modification
 Query ...

Output

☒ Screen ☐ Write to File ☐ Append to File
☐ Print Header ☐ Comma Separator ☐ Trailing Pipe
 ...

☒ Search

Ready BMI Total tasks: 0

The **DFexport** tool window has the following features:

- The top of the main **DFexport** window has a title bar that includes the user name, and study number and name. This is the standard for all DataFax tools.
- A menu tool bar is located just below the title bar.
- A section in which retrieval criteria can be saved as named tasks appears at the top of the window.
- The Selections section, used to specify retrieval criteria and output options, is divided into the following parts:

Plates

Date Format

Options

Lost Code

Headers

Fields

Data Filters

Output

- Three action buttons: Find, Clear Output, and Run sit between the Selections selection and the output window.
- The output window, which is blank when the tool starts, appears at the bottom of the **DFexport** interface.
- Below the output window is a two panel message bar. The first panel displays **DFexport** activity while the second panel displays other messages. Between the two are the filter indicators B, M, and I (see Pre-Defined Filters).

11.2.4. Minimum specifications

Under the Selections section of the dialog first proceed down the left side to select a plate, the output format for dates, the output option for check and choice fields (codes or labels), and the code to appear in the data fields of records that have been registered in the lost data log. If you want to see all data fields and all database records for the selected plate this is all you need to specify. Next, select the Run button and the results will appear in spread-sheet format in the output window.

11.2.5. To narrow your search

To restrict the output to specified data fields and records meeting specified criteria, proceed down the right side of the dialog to select data fields and filters. Data fields can be specified using the field editor dialog invoked using the button to the right of the field list window. Users familiar with **DFexport.rpc** options can enter specifications directly in the fields provided. Each data retrieval option corresponds to exactly one argument of the **DFexport.rpc** command-line program. Any setting that is left blank or unchecked will evaluate to the default value of **DFexport.rpc**. If any data filters are specified, only records which meet all of the selection criteria combined across all of the filters will be retrieved from the study database. The only data selection feature which is not part of **DFexport.rpc** is the query option which works like the retrieval by variables dialog from **DFvalidate**. The query builder dialog can be invoked by clicking the button to the right of the Query text window.

11.2.6. Saving a retrieval

The current **DFexport** settings can be saved at any time as a named user task. Each user's task list is loaded automatically when **DFexport** starts. Thus frequently repeated retrievals can be re-run with little effort. Selecting a task from the task list loads that task's retrieval criteria into the Selections sections of the dialog. Then clicking Run executes **DFexport.rpc** with the current retrieval selections, and the data retrieved from the study database are displayed in the output window at the bottom of the screen.

11.2.7. Output options

The retrieved data records can be written to the output window, an output file, or both. Data written to the output window is presented in spreadsheet format with each row corresponding to a data record and each column corresponding to a data field on the selected study plate. The output can be written or appended to a file in either the standard DataFax format (with fields delimited by |), or in Comma Separated Variable (CSV) format. The results in the output window can be saved to a file using the Save Output option from the File menu at the top of the screen.

11.2.8. Searching the output

The data values in the output window can not be edited in any way, but the rows of the table can be sorted on any column by clicking the variable name in the column title, and the entire table can be searched by entering a string in the Search text field and then clicking the Find button.

11.2.9. Error handling

The error checking performed by the **DFexport.rpc** command line program is also available in **DFexport**. If illegal values or incorrect syntax are encountered, **DFexport** displays the **DFexport.rpc** error or warning message in a pop-up dialog.

11.3. DFexport Tasks

Like the Validation tool, **DFexport** allows users to save their export settings as named user-specific tasks. Each users task list is saved in the `$STUDY_DIR/lib` directory with the name `.DFexport.username`, where `username` is the name of the user who created the task list. Permissions on this file are set to 600. Thus only `username` will be able to read/write this file. The format of `.DFexport.username` is plain text and can be edited using a text editor. However, this is not recommended as editing the file through the **DFexport** tool itself is straightforward and less error-prone. Export tasks are automatically loaded from the task file into the **DFexport** window when the tool is started.

11.3.1. Create an Export Task

To create a user-defined export task:

1. **Set the desired export criteria.** At a minimum a plate must be selected. The other selections in the left side of the panel: date format, options, and lost codes should also be reviewed. All other specifications are optional.
2. **Enter a unique task name in the Tasks Name field.** In the example below, the task name is **Eligibility**.
3. **Select Add...** The export criteria are saved and the task name appears in the task list window.

Figure 11.2. Task Eligibility is added to the scrolling list of user tasks.

The screenshot shows the DFexport application window titled "DFexport [valid1] 254 - Acceptance Test Study". The window has a menu bar with options: File, Edit, Search, Sort, Filter, Misc, Help. Below the menu bar is a "Tasks" section with a "Name" field containing "Eligibility" and buttons for "Add >>", "Delete", and a list of tasks including "Eligibility", "MedCode", and "Task Report". Below the "Tasks" section is a "Selections" section with a "Plates" list containing "000 - New Data Records", "001 - Blood Pressure Screening Visits" (highlighted), "002 - Patient Entry Form", "003 - Medical History", "004 - Current Medications", "005 - Monthly Follow-up Form", and "006 - Patient Evaluation Report". To the right of the "Plates" list is a "Fields" section with a checkbox for "Key [id, plate, visit, status, validation level]" and a "Field Number" field containing "27,28,29,30,31,32". Below the "Fields" section is a "Data Filters" section with fields for "Status", "Validation", "Center", "Patient", "Visit", "Creation", "Modification", and "Query". Below the "Data Filters" section is an "Output" section with radio buttons for "Screen", "Write to File", and "Append to File", and checkboxes for "Print Header", "Comma Separator", and "Trailing Pipe". At the bottom of the window is a status bar with "Ready", "BMI", and "Total tasks: 3".

There is no limit to the number of export tasks that can be defined. Tasks appear in the scrolling task list in alphabetical order. To select a defined task, simply select the task name from the list. The **DFexport** window will then be populated with the settings defined for that task. Selecting a different task will replace the settings with those for the new task. Selecting a different plate clears the current settings for data fields and data filters. All other settings remain unchanged.

11.3.2. Modify an Export Task

To modify an existing export task:

1. **Select the task from the task list.** Single click the desired selection. The defined export criteria will populate the **DFexport** window.
2. **Modify the export criteria.** Any of the current export criteria can be modified.
3. **Select Add... to replace the previous criteria with the current modifications.** The new export criteria will be saved, redefining the current task name.

11.3.3. Delete an Export Task

To delete an existing export task:

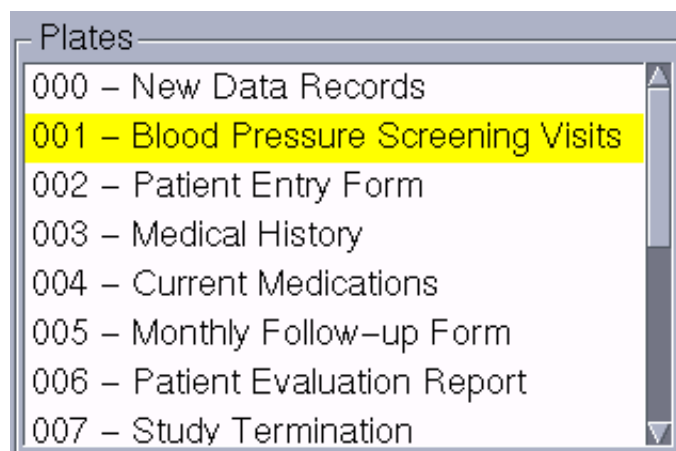
1. **Select the task from the task list.** The defined export criteria will populate the **DFexport** window.
2. **Select Delete.** The name of the export task will disappear from the task list, however the settings will not change. If the Add... button is selected immediately following deletion, the named task will be put back in the task list.

All changes to export tasks are saved when you exit the **DFexport** tool.

11.4. Plate Selection

All plates defined for the current study are displayed in ascending order by plate number in the Plates window as shown below. A description of the plate (as defined in the study schema) follows each plate number.

Figure 11.3. The Plate selection list



A plate is chosen by a single click on the desired entry. Only one plate selection can be made. It is not possible to retrieve and display data fields from multiple plates. Changing the plate selection will clear the Fields and Data Filters text fields of the interface. All other settings remain unchanged.

In addition to user defined plates, corresponding to the study's unique CRF pages, the plate list also contains an entry for plate 000 labeled New Data Records. This is the new record queue containing the ICR results for all CRF pages that have not yet been reviewed and committed to the study database. When records are retrieved from this plate the first 7 data fields (i.e. record status to patient ID number)

are displayed in the first 7 columns of the output window, and the remaining fields are shown as one string of '|' delimited values in the 8th column, under the heading New Data. However, if Calendar or Julian date format is selected, or the Decode Coded Fields option is selected, the New Data column is not displayed because these requests can not be honored for what are as yet unvalidated data records.

11.5. Field Selection

The data fields to be exported are specified in the Fields section of the **DFexport** tool. Field selection is not a requirement and if not specified, all fields on the selected plate will be exported. The data fields to be exported may be specified using one of the following mutually exclusive options:

- keys only (**DFexport.rpc** command-line option `-k`)
- field number (**DFexport.rpc** `-f` option)
- unique name (**DFexport.rpc** `-U` option)
- generic name (**DFexport.rpc** `-G` option)

11.5.1. Exporting Key Fields

If the Key toggle button is enabled, the key fields: patient ID, plate number, visit/sequence number, record status, and record validation level are output in that order, as shown in the following example. This corresponds to using the `-k` option in **DFexport.rpc**. Other data fields can not be output when this option is selected.

Figure 11.4. Key fields for plate 1 are exported to the output screen.

DFexport [valid1] 254 – Acceptance Test Study

File Edit Search Sort Filter Misc Help

Tasks

Name Eligibility Add >> Eligibility
Delete MedCode

Selections

Plates

001 – Blood Pressure Screening Visits
002 – Patient Entry Form
003 – Medical History
004 – Current Medications
005 – Monthly Follow-up Form
006 – Patient Evaluation Report
007 – Study Termination

Date Format

☒ Default ☐ Calendar ☐ Julian

Options

☐ Decode Coded Field ☐ Quiet Mode

Lost Code

☒ Use Default Code [*]
☐ Use

Headers

Fields

☒ Key [id, plate, visit, status, validation level]
Field Number

Data Filters

Status
Validation
Center
Patient
Visit
Creation
Modification
Query

Output

☒ Screen ☐ Write to File ☐ Append to File
☐ Print Header ☐ Comma Separator ☐ Trailing Pipe

Search Find Clear Output Run

	ID001	DFplate1	DFseq1	DFstatus1	DFvalid1
1	10001	1	0	2	1
2	10002	1	0	1	1
3	20025	1	0	0	7
4	20027	1	0	1	1
5	20028	1	0	2	1
6	20030	1	0	2	1
7	99001	1	0	2	1

Plate 001: total 11 records exported BMI Total tasks: 3

In this example the status of each record is shown using its numeric code. For all coded fields, the numeric code is output unless the Decode Coded Field button is enabled, in which case the code label is displayed in place of the numeric code.

11.5.2. Exporting by Field Number

Exporting by field number is equivalent to executing **DFexport.rpc** from the command-line using the **-f** option. When Field Number is selected from the drop-down list in the Fields section of the dialog, field selections can be made by entering field numbers directly into the text field or by clicking the ellipsis button

to the right of the text field to invoke the Field Editor dialog.

11.5.3. Exporting by Field Name

Export by field name is equivalent to executing **DFexport.rpc** from the command-line using the **-U** option for unique name or **-G** option for generic name. Export by field name is performed by selecting either Unique Name or Generic Name from the drop-down list of field selection options in the Fields section of the **DFexport** dialog. When unique or generic name is chosen, field names or numbers can be entered directly into the text field or by clicking on the ellipsis button to the right of the text field to invoke the Field Editor dialog. The choice of generic or unique name determines which names are written to the field list from this dialog, and which names are used as the column titles in the output window.

11.5.4. The Field Editor Dialog

The Field Editor dialog is invoked by selecting the ellipsis button to the right of the field list.

Figure 11.5. The Field Editor dialog.

DFexport – Field Editor

Study: 254 – Acceptance Test Study
 Plate: 002 – Patient Entry Form

☐ Select all fields ☐ Deselect all fields ☐ Constant value before field

	Field	Generic Name	Unique Name	Type	Field Modifier	Constant
<input checked="" type="checkbox"/>	1	DFSTATUS	DFstatus2	choice		
<input type="checkbox"/>	2	DFVALID	DFvalid2	int		
<input type="checkbox"/>	3	DFRASTER	DFraster2	string		
<input type="checkbox"/>	4	DFSTUDY	DFstudy2	int		
<input type="checkbox"/>	5	DFPLATE	DFplate2	int		
<input type="checkbox"/>	6	DFSEQ	DFseq2	int		
<input type="checkbox"/>	7	ID	ID002	int		
<input type="checkbox"/>	8	PINIT	PINIT002	string		
<input type="checkbox"/>	9	EDATE	EDATE	date		
<input type="checkbox"/>	10	MEDCODE	MEDCODE	int		
<input type="checkbox"/>	11	BDATE	BDATE	date		
<input type="checkbox"/>	12	WTKG	WTKG	int		
<input type="checkbox"/>	13	WTLB	WTLB	int		
<input type="checkbox"/>	14	HTCM	HTCM	int		
<input type="checkbox"/>	15	HTIN	HTIN	int		
<input type="checkbox"/>	16	PULSE	PULSE	int		
<input type="checkbox"/>	17	SMOKE	SMOKE	choice		
<input type="checkbox"/>	18	EVERCISE	EVERCISE	choice		

☐ ☒ ☐

When the Field Editor dialog is opened, all fields for the selected plate are listed in the order defined in the study schema, and the current field settings (if any) are shown to be active. The following operations can be performed to manipulate field selection criteria within this dialog.

- **Select fields for export.** To select individual fields use the toggle button to the left of each field. Or use the Select all fields and Deselect all fields toggle buttons at the top of the dialog.
- **Change field output order.** Select the desired field then move it up or down in the table using the arrows at the bottom left of the dialog.
- **Duplicate fields.** Select the desired field and click the duplicate (plus sign) button. This is useful if you want to output both code values and code labels for the same field.
- **Add a field modifier.** Select a field and enter the desired modifier into the Field Modifier column. Field modifiers are the same as those used in **DFexport.rpc** and include the date modifiers :c for calendar format, :j for julian format, and :o for original study schema format, plus the decoding modifier :d to output code labels.

- **Include constant values.** A constant value can be inserted before or after any of the exported data fields. Select a field and enter the desired constant value into the Constant Value column. Constant values must be enclosed within single quotes, e.g., 'P2'. It is necessary to press the `Enter` key to save the constant value. By default constant values are inserted after the data field. To insert the constant value before the data field use the Constant value before field toggle button at the top of the dialog.
- **Apply your specifications.** Select OK to apply your field specifications. They will then appear in the Field list and the Field Editor dialog will close.
- **Clear all field specifications.** Select Reset to re-initialize the Field Editor dialog. This clears all settings including field level modifiers and constant values.
- **Undo changes.** Select Cancel to discard all changes and close the Field Editor dialog.

11.6. Data Filters

This section of the **DFexport** window is used to narrow the output to include only those records with specified values in specified fields. It includes the same options available in **DFexport.rpc** for record status, validation level, center ID, patient ID, visit number, and record creation and modification dates. It also includes the ability to build a logical data query using one or more of the data fields on the plate, and the ability to filter data records that contain one or more of blank required fields, missing value codes or illegal values. If more than one filter is specified, only those records that contain fields that meet all the specified criteria will be output. Thus if you specify a filter that is not met by any of the fields on database records (e.g. '99' in the center filter, when center 99 is not defined for the study) no output will be generated, even though there may exist data records that meet some of the other specified filters. Data filters and their equivalent **DFexport.rpc** options are described in the sections that follow.

11.6.1. Status (*DFexport.rpc* option -s)

The Status filter is used to select data records by record status. The status keywords: `clean`, `dirty`, `error`, `CLEAN`, `DIRTY`, `ERROR`, `lost`, `primary`, `secondary`, `all` are valid specifications. One or more of these keywords, delimited by a comma (,), can be specified.

Record status may be specified by entering the desired keyword in the text field provided, or by selecting the ellipsis button to the right of the text field to invoke the Status Editor dialog. In the dialog, the desired record or QC note status (for plate 511) can be chosen using the toggle button to the left of the rows. Alternatively, All statuses, Primary records, or Secondary records can be selected by using the buttons at the top of the dialog. In the example below, the Primary records button was enabled, resulting in all primary record statuses being selected for export.

Figure 11.6. The Status Editor dialog.

DFexport – Status Editor

Study: 254 – Acceptance Test Study
Plate: 002 – Patient Entry Form

☐ All statuses
☒ Primary records
☐ Secondary records

	Record Status	QC Note Status
<input checked="" type="checkbox"/>	clean	new
<input checked="" type="checkbox"/>	dirty	in QC report
<input checked="" type="checkbox"/>	error	resolved NA
<input type="checkbox"/>	CLEAN	resolved irrelevant
<input type="checkbox"/>	DIRTY	resolved corrected
<input type="checkbox"/>	ERROR	in sent QC report
<input type="checkbox"/>	lost	

If status `lost` is selected, any records registered in the lost data log will be included in the output. But since these records do not have data values for the user defined fields on the plate, these fields are filled with the selected lost code, which can be, the DataFax missing value code (*), a user-specified missing value code or a blank (see Lost Data Codes). In the following example records 2, 6 and 7 are lost records.

Figure 11.7. Output obtained by exporting `lost` and `clean` records for Plate 1.

DFexport [valid1] 254 – Acceptance Test Study

File Edit Search Sort Filter Misc Help

Tasks

Name Eligibility Add >> Eligibility
Delete MedCode

Selections

Plates

001 – Blood Pressure Screening Visits
002 – Patient Entry Form
003 – Medical History
004 – Current Medications
005 – Monthly Follow-up Form
006 – Patient Evaluation Report
007 – Study Termination

Date Format

☒ Default ☐ Calendar ☐ Julian

Options

☒ Decode Coded Field ☐ Quiet Mode

Lost Code

☒ Use Default Code [*]
☐ Use

Headers

Fields

☐ Key [id, plate, visit, status, validation level]
Field Number 1,7-NF

Data Filters

Status clean,lost
Validation
Center
Patient
Visit
Creation
Modification
Query

Output

☒ Screen ☐ Write to File ☐ Append to File
☐ Print Header ☐ Comma Separator ☐ Trailing Pipe

Search Find Clear Output Run

	DFstatus1	ID001	PINIT001	AGE	SEX	RACE	RACEOTH	S1 DATE	S1SBP1	S1
1	clean	10002	PPP	40	male	caucasian		04/05/04	111	111
2	lost	20025	*	*	*	*	*	*	*	*
3	clean	20027	TAB	60	male	asian		03/05/05	111	100
4	clean	99002	JAM	64	male	caucasian		04/01/97	168	096
5	clean	99004	TRB	70	male	asian		01/06/97	172	099
6	lost	99010	*	*	*	*	*	*	*	*
7	lost	99067	*	*	*	*	*	*	*	*

Plate 001: total 7 records exported BMI Total tasks: 3

To clear all settings in the Status Editor dialog, select Reset. To exit the Status Editor, without saving your changes select Cancel. To apply the export criteria displayed in the Status Editor, select OK.

11.6.2. Validation (DFexport.rpc option -v)

The Validation filter is used to select data records by validation level. Any level or range of levels between 0-7 can be specified. However, validation level 0 applies only to new records. A comma delimiter is used to separate list components, e.g. 1-3,6.

11.6.3. Center (*DFexport.rpc option -n*)

Data record selection may be performed by center using any list of center numbers and ranges that are valid for the study. Center lists must be comma (,) delimited, e.g., 1-9,20,30. The center number(s) entered must correspond to those defined in the centers database (**DFcenters**). It is not valid to specify both center and patient numbers for the same export task. Doing so will generate an error message.

11.6.4. Patient (*DFexport.rpc option -I*)

Data records may be selected by patient ID number by specifying a list of ID numbers and ranges. A comma delimiter (,) must be used to separate the list components, e.g., 20001,30001,99001-99020. It is not valid to specify both patient and center numbers for the same export task. Doing so will generate an error message.

11.6.5. Visit (*DFexport.rpc option -v*)

Data records can be selected by the DataFax visit or sequence number found in field 6 of each data record. Valid DataFax visit/sequence numbers include any integer in the range of 0-65535 and are not required to be defined in the study visit map (**DFvisit_map**). As for center and patient numbers, a single visit, range of visits, or a combination thereof, can be specified using a comma delimiter.

11.6.6. Creation date (*DFexport.rpc option -c*)

Data records can be selected by their creation date. These dates are assigned by DataFax at the time a record is first committed to the database with a validation level greater than zero, and are stored at the end of each data record in the second last data field. If specified, dates must be entered in the DataFax default format of *yy/mm/dd*. A comma delimited list comprised of one or more dates and/or ranges may be specified. This filter is not available for, new records awaiting data entry (plate 000), QC notes (plate 511) or reason records (plate 510).

11.6.7. Modification date (*DFexport.rpc option -m*)

Data records can be selected by the date on which they were last modified. This date is updated by DataFax when a record is committed to the database if any of the user defined data fields have been changed, and is stored at the end of each data record in the last data field. As for Creation dates, Modification dates must be specified in the DataFax default format of *yy/mm/dd*. This filter is not available for, new records awaiting data entry (plate 000), QC notes (plate 511) or reason records (plate 510).

11.6.8. Query

Data selection by Query is similar to Validation tool retrieval By Variables (see *DataFax User Guide, By Variables*). This filter can be used for any plate except for new records awaiting data entry (plate 000). A query can be entered by typing in the Query text field, or using the Query Editor dialog which is opened by clicking on the ellipsis button to the right of this field.

An existing query, visible in the Query text field, will be removed if a new plate is selected, or if the field list, date format, or decoding option is changed. Thus make sure these specifications have been entered before you start creating a query. Also, it is important to note that a query can only be created using data fields included in the current list of fields to be retrieved. Thus for example, to select patients over 80 years old, the age data field must appear in the current list of fields to be exported from the database.

Figure 11.8. The Query Editor dialog.

Study: 254 – Acceptance Test Study
Plate: 001 – Blood Pressure Screening Visits

Query

Field	Name	Type	Format
1	DFstatus1	choice (code)	
2	DFvalid1	int	
3	DFraster1	string	
4	DFstudy1	int	
5	DFplate1	int	
6	DFseq1	int	
7	ID001	int	
8	PINIT001	string	
9	AGE	int	
10	SEX	choice (code)	
11	RACE	choice (code)	
12	RACEOTH	string	
13	S1DATE	date	dd/mm/yy
14	S1SBP1	int	

Symbols

- (
-)
- == EQUALS
- != NOT EQUAL
- ~ CONTAINS
- >= GREATER EQUAL
- > GREATER
- <= LESS EQUAL
- < LESS
- || OR

Codes

- 0 lost
- 1 clean
- 2 dirty
- 3 error

Clear OK Cancel

A query can be created in the Query Editor dialog by typing it directly into the Query text field, or by selecting Variables, Symbols and Codes as needed from the dialog. Only those variables listed in the Field selection list will be displayed as only these variables can be used to build a query. When your query is complete select the OK button. This will write your query to the Query text field under Data Filters and close the Query Editor dialog.

The following rules apply when specifying data values in a query:

- Values for string fields must be specified in double quotes. An empty string (blank value) is represented by two consecutive double quotes with no space between them.
- Dates must also be enclosed in double quotes using the format shown for the variable in the Query Editor dialog.
- The record creation and modification fields may be used in a query for all plates except New records (plate 000), QC notes (plate 511) and reason for data change (plate 510). These fields have format `yy/mm/dd hh:mi:ss`, where `yy/mm/dd` is the date in year/month/day format and `hh:mi:ss` is the time in 24-hour:minutes:seconds format. Again values must be enclosed in double quotes. The time

component is optional. If omitted 00:00:00 is assumed.

- For numeric values leading zeros are not significant.
- Values for string fields are case sensitive.
- For coded fields, values must be specified as numeric codes if codes are being output, and as text labels (in double quotes) if labels are being output. The current requirement for each field can be determined by checking the Variable Type column in the Query Editor dialog.

The following examples illustrate valid query syntax.

Example 11.1. List records in which age lies between 60 and 69, inclusive

```
$(AGE) ≥ 60 && $(AGE) ≤ 69
```

Example 11.2. List records in which age is blank or missing

```
$(AGE) == "" || $(AGE) == "*" || $(AGE) == "X"
```

This example includes a test for an empty field and for 2 different missing value codes, the DataFax default missing value code *, and a user defined missing code "X". Missing value codes, defined in the study missing values map, must be specified explicitly in queries; as there is no single function that can be used to test for all missing value codes.

Example 11.3. List records in which screening occurred before Dec. 5,2004

```
$(S1DATE_001) < "05/12/04"
```

Example 11.4. List only clean and dirty records

```
$(DFstatus1) == "clean" || $(DFstatus1) == "dirty"
```

Labels have been specified instead of the corresponding codes because the decode option is selected. This can be verified by noting that Variable Type is displayed as 'choice (label)', and not 'choice (code)', in the Query Editor dialog.

Example 11.5. List males over 60 and females over 70

```
($(SEX_001) == "male" && $(AGE) > 60) || ($(SEX_001) == "female" && $(AGE) > 70)
```

Not the use of brackets to group the different logical tests for men and women.

11.6.9. Pre-Defined Filters

DFexport includes pre-defined filters for selecting records that contain: blank required fields, fields with missing value codes, and fields with illegal values. These filters can be applied to all fields of user-defined plates, QC plates (plate 511), and reason for change plates (plate 510), but not to new records awaiting data entry (plate 0). Just as for other export criteria, the settings for pre-defined filters can be saved to a user **DFexport** retrieval task.

There are 3 types of pre-defined filters:

1. **Blank Required Fields.** When this filter is enabled, only those records that contain one or more blank required fields are exported. To fulfill this criteria, fields must be defined as required in the study schema and contain either
 - no value
 - the code or label corresponding to "blank" for choice/check fields
 - a value of 0 for dates that are converted to julian format

2. **Missing Value.** When this filter is enabled, only those records that contain one or more DataFax missing value codes are exported. Missing value codes must be defined in the `DFmissing_map` file. The following behavior is observed with respect to `DFmissing_map`:
 - if `DFmissing_map` does not exist, the DataFax default missing value code (*) is used
 - if `DFmissing_map` exists but is empty, no missing value codes are defined for the study. The default code of * is no longer considered to be a missing value code and instead is treated as a data value
 - if `DFmissing_map` contains one or more entries, the defined missing value codes are used

3. **Illegal Value.** When this filter is enabled, only those records that contain one or more illegal values are exported. The following are considered to be illegal values:
 - a value outside of the field's legal range as defined in the study schema
 - a partial date value that cannot be imputed
 - a value of -1 for dates that are converted to julian format
 - a string value that is longer than the field's maximum store value as defined in the study schema

Constant values are always considered to be legal values. If string splitting is used to split a long string into multiple fields, only the first field is tested for legality and the remaining fields are treated as constant values.

To enable one or more of the pre-defined filters, select the desired filter item(s) from the Filter menu. To disable an option, de-select it from the Filter list. When printing the export output to the screen, you may also choose to enable the filter color settings. Colors for each filter type can be enabled/disabled by selecting the desired filter from the Misc menu. When colors are enabled, the filtered data will be displayed in the output window with the following colors:

- Blank Required Fields = yellow
- Missing Value = green
- Illegal Value = red

Filter indicators and status are located in the middle of the status bar at the bottom of the main **DFexport** tool window. Filter indicators are represented by uppercase letters: **B** (Blank Required Fields), **M** (Missing Value), and **I** (Illegal Value). If a filter indicator letter appears grayed-out, that particular filter is disabled in the Filter menu. If it appears black, that particular filter is enabled. When a filter indicator appears inside a colored background, its filter color setting is enabled. A white background means the filter color setting is disabled.

11.7. Date Formats

An output date format is specified by enabling the desired radio button in the Date Format section of the main **DFexport** dialog. Three options are available: Default (the date format defined for each date field in the study schema), Calendar (the date format defined in the study schema but with year expanded to 4 digits) and Julian (number of days since the inception of the Julian calendar in 4713 BC). All date fields will be displayed in the selected format except for dates that have been assigned a field level modifier (: o for original schema format, : c for calendar format, or : j for julian format).

The example below shows the output obtained by exporting the 2 unmodified blood pressure screening dates from plate 1 in calendar format.

Example 11.6. Blood pressure screening dates exported in calendar format.

DFexport [valid1] 254 – Acceptance Test Study

File Edit Search Sort Filter Misc Help

Tasks

Name: Eligibility Add >> Eligibility Delete MedCode

Selections

Plates

001 – Blood Pressure Screening Visits
 002 – Patient Entry Form
 003 – Medical History
 004 – Current Medications
 005 – Monthly Follow-up Form
 006 – Patient Evaluation Report
 007 – Study Termination

Date Format

☐ Default ☒ Calendar ☐ Julian

Options

☐ Decode Coded Field ☐ Quiet Mode

Lost Code

☒ Use Default Code [*]
☐ Use

Headers

Fields

☐ Key [id, plate, visit, status, validation level]
 Field Number: 13,20

Data Filters

Status
 Validation
 Center
 Patient
 Visit
 Creation
 Modification
 Query

Output

☒ Screen ☐ Write to File ☐ Append to File
☐ Print Header ☐ Comma Separator ☐ Trailing Pipe

Search Find Clear Output Run

	S1 DATE	S2 DATE
1	03/04/2004	04/04/2004
2	04/05/2004	06/05/2004
3	*	*
4	03/05/2005	03/05/2005
5	03/05/2005	07/05/2005
6	19/06/2004	20/06/2004
7	03/03/1997	05/03/1997
8	04/01/1997	08/01/1997

Plate 001: total 11 records exported BMI Total tasks: 3

11.8. Options

11.8.1. Decode Coded Fields

By default choice and check fields are output using the numeric codes defined in the study schema. This can be changed to output the code labels either by enabling the Decode Coded Field button or by using the field level modifier :d. The example below shows the output obtained by choosing the Decode Coded Field option.

Example 11.7. Selecting Decode Coded Field outputs value labels for all choice and check fields.

DFexport [valid1] 254 – Acceptance Test Study

File Edit Search Sort Filter Misc Help

Tasks

Name Eligibility Add >> Eligibility
Delete MedCode
Term Report

Selections

Plates

001 – Blood Pressure Screening Visits
002 – Patient Entry Form
003 – Medical History
004 – Current Medications
005 – Monthly Follow-up Form
006 – Patient Evaluation Report
007 – Study Termination

Fields

☐ Key [id, plate, visit, status, validation level]
Field Number 1,4,5,6,7,27,28,29,30,31,32 ...

Data Filters

Status
Validation
Center
Patient
Visit
Creation
Modification
Query

Date Format

☐ Default ☒ Calendar ☐ Julian

Options

☒ Decode Coded Field ☐ Quiet Mode

Lost Code

☒ Use Default Code [*]
☐ Use

Headers

Output

☒ Screen ☐ Write to File ☐ Append to File
☐ Print Header ☐ Comma Separator ☐ Trailing Pipe

Search Find Clear Output Run

	DFstatus1	DFstudy1	DFplate1	DFseq1	ID001	ELIG1	ELIG2	ELIG3	ELIG4	ELIG5	EL
1	dirty	254	1	0	10001	Yes	Yes	No	No	Yes	Ye
2	clean	254	1	0	10002	Yes	Yes	No	No	Yes	Ye
3	lost	254	1	0	20025	*	*	*	*	*	*
4	clean	254	1	0	20027	Yes	Yes	No	No	Yes	No
5	dirty	254	1	0	20028	Yes	Yes	No	No	Yes	No
6	dirty	254	1	0	99001	Yes	Yes	No	No	Yes	Ye
7	clean	254	1	0	99002	Yes	Yes	No	No	Yes	Ye

Plate 001: total 10 records exported BMI Total tasks: 3

11.8.2. Quiet Mode

To enable quiet mode, select the Quiet Mode button. Executing **DFexport** in quiet mode prevents warning messages from being displayed. Quiet mode has no effect on error messages.

11.9. Lost Data Codes

The Lost Code section in the **DFexport** dialog is used to define a code to be inserted into the data fields of records that have been registered in the lost data log, when these records are included in the output (as they are unless specifically excluded using a status field filter). By default, the Use Default code (*) option is enabled when **DFexport** is opened. This automatically inserts the DataFax default missing value code of * into all user defined data fields of lost data records. To specify a different missing value code, enable the Use button and enter the desired code in the field provided. If the Use button is enabled but the code field is left blank, the lost record data fields will also be left blank.



Note

DFexport always displays lost records with the same field list as all other records. Lost data records can only be exported in the lost record format, which includes the lost reason code and description, by running **DFexport.rpc** from the command-line.

11.10. Data Output Options

The Output section of the **DFexport** dialog is used to specify the output destination (Screen, Write to File, Append to File) and formatting (Print Header, Comma Separator, Trailing Pipe). Only one output location can be specified at a time. Output formatting can include none or one or more of the 3 options represented by the square toggle buttons. Each of the Output options are described below.

11.10.1. Screen

Enabling the Screen button will send the output from **DFexport.rpc** to the output window in the lower half of the **DFexport** dialog. Screen is the default output option and is automatically enabled each time **DFexport** is opened.

11.10.2. Write to File

Output can be written to a specified filename by selecting the Write to File button. When Write to File is enabled, the text field in the output options section is also enabled. The complete path name of the output file must be entered in this field, either directly or by using the Select File dialog invoked by selecting the ellipsis button located to the right of the text field. The study's `/work` directory is opened by default when this dialog is opened. This can be changed by navigating to a different directory. If changed, **DFexport** will remember the new directory and display it the next time the dialog is opened in the same **DFexport** session.

11.10.3. Append to File

The Append to File option allows you to add the output to the end of an existing file, rather than overwriting it.

11.10.4. Print Header

If the field list is constructed using field number or unique variable names, unique variable names are displayed in the header at the top of each column of the output window. If the field list is constructed using generic variable names, these names are used instead. The variable name header is always displayed when output is written to the screen, regardless of whether or not the Print Header button is enabled.

For new records awaiting data entry (plate 0), variable names differ from plate to plate and can thus not be used as headers. However, fields 1-7 are fixed DataFax key fields and their names are used as headers for new records. All fields following field 7 (patient ID), are displayed in one column of the output table with the header New Data.

For quality control notes (plate 511) and reason for change records (plate 510), headers are displayed using the fixed DataFax variable names defined in the study schema for these plates.

If constant values are included in the output, column header names can be specified for these values in the Header text window, which becomes active when the Print Header button is enabled. A list of names, separated by space or commas, can be entered to assign a header to each constant value (from left to right in the field list). If no header is specified the constant value itself is used as the column name.

When exported data is written or appended to a file instead of being output to the screen, headers are not automatically included in the output. If you want headers to appear as the first record in the output file, the Print Header option must be enabled.

11.10.5. Comma Separator

By default, the pipe (|) character is used to separate each field when exported records are written to a file. This includes header records. The "|" character may be replaced by a comma by selecting the Comma Separator option. This results in data and header records being output in comma separated variable format. This option only effects output written to a file.

11.10.6. Trailing Pipe

A trailing pipe delimiter can be inserted at the end of each exported data record to maintain the original DataFax record format so that exported records may be re-imported into the database using the **DFimport.rpc** program. To include a trailing pipe delimiter at the end of each record enable the Trailing Pipe button. This option only effects output written to a file.



Note

DataFax does not expect a trailing pipe delimiter to be present in quality control note (plate 511) or reason for change (plate 510) data records. Hence, the Trailing Pipe option should not be enabled when exporting data from these plates.

11.11. Output Window

The output window at the bottom of the main **DFexport** window is populated if and only if, the Output selection is set to Screen. The output window is a read-only table widget. The label appearing at the top of each column represent either unique variable names (default) or generic variable names. Each row in the table represents a single record in the database with each cell representing a data field. Each exported record is numbered in the first column of the table.

The width of the columns are adjusted to fully display the data value and column header for each field, except for very long text fields which are only displayed to a maximum of half the width of the display window. However, if the last field in the record is a long text field, the last column of the table will be adjusted to display the longest data value. Columns in the output table may be resized by clicking and dragging on the boundary between column headers.

The entire Selection section of the dialog can be collapsed by clicking the arrow button located to the left of the Search text field. This maximizes the space available for the output window.

11.12. Actions

11.12.1. Run

Selecting the Run button executes **DFexport.rpc** with the specifications currently set in the **DFexport** dialog. If the output option is set to Screen, any data currently displayed in the output window will be cleared and replaced by the new output.

11.12.2. Clear Output

To clear the output window, select Clear Output. This clears the output window only and will not alter any of the current **DFexport** settings.

11.12.3. Search and Find

The results displayed in the output window may be searched for a specified text string. To do this, enter the desired string in the Search text field located above the output window and select the Find button. By default, the search will match on any portion of the field value. The search begins from the current cell in the output table. A cell is made current by clicking on it. The first match that is found becomes the new current cell and its record is highlighted in the table. To continue the search, select Find again. If no further matches are found, the last match remains as the current cell and you will be given the option of beginning the search again from the first cell of the table.

Search options, located under Search in the menu bar at the top of the screen are described below in Title Bar Menus.

11.12.4. Sort

The rows of the output table can be sorted on any output data field by clicking on the variable name at the top of each column. By default, records are sorted in ascending order, and all values are considered case-sensitive.

Sort options, located under Sort in the menu bar at the top of the screen are described below in Title Bar Menus.

11.13. Title Bar Menus

11.13.1. File

- **Save Output.** The Save Output option is used to save the output displayed in the output window to a computer file. Selecting this option opens the Select File dialog which is used to select an output file. Any formatting specified by the Output options Print Header, Comma Separator, and/or Trailing Pipe, will be used in the saved output even though they are not displayed in the output window itself.
- **Save DataFax Retrieval File.** Records in the output window may be saved to a DataFax Retrieval File (DRF). Selecting Save DataFax Retrieval File from the File menu invokes the Save DataFax Retrieval File dialog. The dialog requires a path/filename for your DRF and allows the specification of file permissions and other options described below.

Include Image ID. When this option is enabled, the record's image ID will be included in the DRF as the 4th field (fields 1, 2, and 3 are patient ID, visit, and plate respectively). This option is not available for QC note (plate 511) and reason for data change (plate 10) records as neither are associated with a record image. If the image ID field has not been included as export criteria for the exported records, this option is also disabled in the Save DataFax Retrieval File dialog.

Write to File. When this option is enabled, a new DRF will be created. If the specified DRF already exists, the existing DRF will be overwritten, and the original owner of the file is retained.

Merge to File. When this option is enabled, the records in the output window will be merged into the specified DRF. If the specified DRF does not exist, a new DRF will be created. After the merge, the contents of the DRF will be sorted in ascending order by patient ID, visit, plate and image ID.



Note

The patient ID, visit and plate key fields must be known and included in the exported records in order to create a DRF. Hence, new records awaiting data entry (plate 0) cannot be saved to a DRF as their key fields are not always known or correct.

- **Exit.** Select Exit to close the **DFexport** tool. If the program completed normally it will exit with status 0, otherwise the exit status will be greater than 0.

11.13.2. Edit

The Edit menu consists of the actions > Undo, Redo, Cut, Copy, and Paste. Each of these actions apply to any editable text field in the **DFexport** dialog (i.e. to Data Filters, Headers, and Search fields). They do not apply to the various radio and selection buttons.

11.13.3. Search

The options available in the Search menu allow you to change the default search algorithm as follows.

- **Match Exact.** This specifies that the search string must match the entire field value. Case sensitivity can be specified along with Match Exact by enabling/disabling the Match Case item. Only one of Match Exact or Match Word can be specified at a time.
- **Match Word.** This specifies that the search string constitutes a word and that only words are to match, e.g. 'head' will only match the word 'head' and not 'headache', 'ahead' or any other string containing 'head'. Case sensitivity can be specified with this search option by enabling/disabling the Match Case item. Only one of Match Word or Match Exact can be specified in a search.
- **Match Case.** With this option enabled the match is case sensitive, and without it upper and lower case are treated identically.
- **Find.** This option is equivalent to the Find button located above the output window.

11.13.4. Sort

The following sort options are available:

- **Descending Order.** If this button is enabled, sorting on the selected output column is performed in descending order. Otherwise ascending sort order is used.
- **Case Insensitive.** If this button is enabled, case is ignored when sorting string values.
- **As String.** If this option is enabled, values are sorted as character strings. If the data type is numeric or date you will generally not want to use this option.

For numeric fields, non-numeric values (i.e. missing value codes and blanks), appear first when sorting in ascending order, and last when sorting in descending order.

Date imputation, as specified in the study schema, is performed before sorting date fields. Date values having the format `username timestamp` such as QC creation/modification dates, are sorted on the date component first, followed by the username component. Invalid dates (i.e. missing value codes, partial dates on fields that do not allow imputation, and nonsensical values for day and month) appear first when sorting in ascending order and last when sorting in descending order.

11.13.5. Filter

The Filter menu allows the selection of records containing:

- Blank Required fields
- Missing Value codes
- Illegal Values

One or more of these filters may be enabled for a single export run (see Pre-Defined Filters).

11.13.6. Misc

The Misc menu allows the user to enable/disable the colors for the filtered fields (Blank Required, Missing Value, Illegal Value) displayed in the output window. Color coding matches the default field color setting in **DFvalidate** (see Pre-Defined Filters).

11.13.7. Help

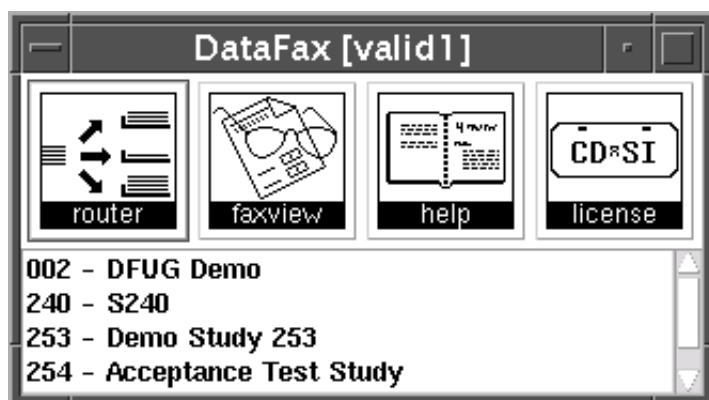
- **Contents.** Selecting Contents displays the documentation for the **DFexport** tool from the DataFax User Guide.
- **On Version.** Selecting On Version displays version and user information for **DFexport** and **DFexport.rpc**.

Chapter 12. Unidentified Fax Router

12.1. Introduction

When DataFax receives pages which it cannot recognize, it sends them to the DataFax router. The Unidentified Fax Router (**DFrouter**) is used to manually review these pages, and send them where they belong. The **DFrouter** is started from the DataFax Tool Box by selecting the Router icon, shown below on the left.

Figure 12.1. DFrouter is started from the DataFax User Tool Box.



Pages arrive in **DFrouter** whenever DataFax is unable to find or read a CRF bar code on the page. This occurs either because there was no bar code (e.g. memos, letters, cover pages), or because something interfered with its ability to read a CRF bar code (e.g. fax noise, a coffee stain).

Without a readable bar code DataFax can't assign the page to the correct study; thus the **DFrouter** must hold all unidentified faxes for all studies. If you are running several studies, the individual(s) who review and dispatch pages from **DFrouter** must be sufficiently familiar with all of them to correctly route memos and letters.

Study CRFs are not difficult to route correctly unless the bar code is missing altogether. In such cases the contents and design of the CRF may be enough to correctly identify it. If not, selecting Context from the Fax can be used to determine where the CRF came from and that site can be asked to re-fax it. Also, DataFax QC reports automatically include notification of missing pages and overdue visits, so required CRFs (defined in the study visit map) can be safely deleted from **DFrouter** if they are too damaged to be legible.

12.2. DFrouter Functions

Through **DFrouter** you will be able to:

1. Read all unidentified pages in **DFrouter** on screen.
2. Get context (determine when the fax arrived and where it came from).
3. Delete pages that are not needed (e.g. fax cover pages, blank pages).
4. Print pages if a paper copy is desired.
5. Send pages to another user who can then read them using the Faxview Tool (**DFfaxview**).
6. Fix any fax transmission problems on CRF pages (flip, rotate, truncate, cut).

7. Identify CRFs (study, plate, visit) and send them to the regular fax input stream.

12.3. Starting DFrouter

Multiple users may simultaneously use **DFrouter**. When a user selects **DFrouter** from the DataFax Tool Box, all pages belonging to a single fax transmission are locked and loaded into that user's **DFrouter** session. Faxes are locked and released as required during user traversal through the record list. As a user moves through the list small icons show whether the record has been processed (routed/deleted/saved/sent to another user) (a red X icon), is currently locked by another user (a lock icon), is currently locked by this user (a exclamation mark icon) or is available (no icon).

A fax transmission is defined by the combination of a four digit year and week number, followed by a four character alphanumeric fax ID, and then three digits representing the fax page number. The locking mechanism is handled by **DFmaster.rpcd**. Fax transmissions are locked and loaded, beginning with the oldest fax transmission that is waiting to be routed.

The example below shows the current **DFrouter** session for user `datafax`. All pages in the router are listed chronologically (newest at the top) in the record list panel to the right of the fax image. Pages locked by this user in the current **DFrouter** session are identified by the green exclamation mark icon. The bottom right of the screen displays information about the fax page image ID and the image resolution of the current page (in pixels).

Figure 12.2. The main DFrouter window.

DFrouter [valid1]

File Page Action Help

04/20/04 10:13 FAX 905 522 7284 CDSI DATAFAX → CDSI DataFax 002

DataFax ICR Test Page Form 1

DataFax #247 Plate #305

Please follow the instructions below, use a dark blue or black pen, and keep each digit inside the box. The examples you provide on this page will be used to help us improve the Intelligent Character Recognition (ICR) component of DataFax 4.0.

CAREFUL: for each of the following 3 rows print the digits 0-9 as clearly and carefully as you can

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

TYPICAL: for each of the following 3 rows print the digits 0-9 at a typical pace

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

RUSHED: for each of the following 3 rows print the digits 0-9 as you would if in a hurry

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

CRF ID

0404.0067002

0404.0082001

0404.0101002

0411.0008001

0411.0008002

0412.0002001

0415.0009001

0415.0008001

0415.000C001

0415.000D001

0416.0001001

0416.000J002

0416.000K022

0416.000N001

0416.000N002

0416.000P001

0417.000S001

0417.000S002

0418.0003001

0418.0003002

0418.0003004

0418.0003006

0418.0003007

0418.0003008

0418.0003009

0418.000D001

0418.000F001

0419.0003001

0419.0004001

0420.000D003

0420.000P001

32 routable pages (0 busy) 0416.000J002 [864x1079]

12.4. The File Menu

This menu contains 2 items: Update Record List and Exit.

12.4.1. Update Record List

Select Update Record List to update the record list with any unidentified fax pages that may have arrived after the current user session began.

12.4.2. Exit

Select Exit from the File to close **DFrouter**.

12.5. The Page Menu

This menu contains a number of functions which allow you to manipulate (rotate, shift, flip, cut, truncate, and reset) the pages that end up in **DFrouter**. If you see a study CRF in **DFrouter** you will need to clean it up using these functions before you identify and route it to the study database.

This menu also contains functions to turn to the next and previous page, print pages and determine where each page came from (context).

12.5.1. *Rotate*

Select Rotate from the Page menu to correct page skew. This is needed if the page signature line at the top of a study CRF is not perfectly horizontal. Page skew usually results from failure to adjust the page guides snugly against the sides of the CRF pages when they are being scanned into the fax machine.

Figure 12.3. Rotate instructions appear at the bottom of the screen upon selecting Rotate from the Page menu.

DFrouter [valid1]

File Page Action Help

ical DataFax Systems, Inc.
Study 254

Example Blood Pressure Trial

DataFax #254 Plate #001 Sequence#000

Patient Initials F M L

Screen 1: Date day month year

Screen 2: Date day month year

BLOOD PRESSURE SCREENING VISITS

Age years Sex Male Female Race Caucasian African American Asian Other, specify:

Reading 1: systolic diastolic mmHg

Reading 2: systolic diastolic mmHg

Reading 3: systolic diastolic mmHg

CRF ID

- 0404.0101002
- 0411.0008001
- 0411.0008002
- 0412.0002001
- 0415.0009001
- 0415.0008001
- 0415.000C001
- 0415.000D001
- 0416.0001001
- 0416.000J002
- 0416.000K022
- 0416.000N001
- 0416.000N002
- 0416.000P001
- 0417.000S001
- 0417.000S002
- 0418.0003001
- 0418.0003002
- 0418.0003004**
- 0418.0003006
- 0418.0003007
- 0418.0003008
- 0418.0003009
- 0418.000D001
- 0418.000F001
- 0419.0003001
- 0419.0004001
- 0420.000D003
- 0420.000P001
- 0420.000T001

Eligibility Criteria

Yes No

Click left mouse button on left end of horizontal bar, right button to Cancel 32 routable pages (0 busy) 0418.0003004 [864x1096]

Instructions for the operation appear in the status line at the bottom of the **DFrouter** window. Click the left mouse button on the top of the left end of the horizontal signature line, or click the right mouse button to cancel the operation. After clicking the left mouse button once, click the left mouse button on the top right end of the signature line to complete the operation or click the right mouse button to cancel. Once the operation is complete, the screen will update to show the realigned image. You can undo a rotation by selecting Reset from the Page menu.

12.5.2. Shift

Shift is used to register the upper-left corner of the page signature line found at the top of all DataFax CRFs. The upper left corner of this line is used as the CRF reference point for DataFax ICR and thus must be registered before the page is identified and routed to the study database. If you fail to do this the ICR software will look for data in the wrong places on the CRF page and will thus fail to find data fields and will leave them blank.

Figure 12.4. Shift instructions appear at the bottom of the screen when Shift is selected from the Page menu.

DFrouter [valid1]

File Page Action Help

04/20/04 10:13 FAX 905 522 7284 CDSI DATAFAX → CDSI DataFax 002

DataFax ICR Test Page Form 1

DataFax #247 Plate #305

Please follow the instructions below, use a dark blue or black pen, and keep each digit inside the box. The examples you provide on this page will be used to help us improve the Intelligent Character Recognition (ICR) component of DataFax 4.0.

CAREFUL: for each of the following 3 rows print the digits 0-9 as clearly and carefully as you can

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

TYPICAL: for each of the following 3 rows print the digits 0-9 at a typical pace

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

RUSHED: for each of the following 3 rows print the digits 0-9 as you would if in a hurry

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

CRF ID

- 0404.0101002
- 0411.0008001
- 0411.0008002
- 0412.0002001
- 0415.0009001
- 0415.0008001
- 0415.000C001
- 0415.000D001
- 0416.0001001
- 0416.000J002**
- 0416.000K022
- 0416.000N001
- 0416.000N002
- 0416.000P001
- 0417.000S001
- 0417.000S002
- 0418.0003001
- 0418.0003002
- 0418.0003004
- 0418.0003006
- 0418.0003007
- 0418.0003008
- 0418.0003009
- 0418.000D001
- 0418.000F001
- 0419.0003001
- 0419.0004001
- 0420.000D003
- 0420.000P001
- 0420.000T001

Click left mouse button on top of horizontal bar, right button to Cancel 32 routable pages (0 busy) 0416.000J002 [864x1079]

Instructions for the operation appear in the status line at the bottom of the **DFrouter** window. Click the left mouse button on the top edge of the horizontal signature line, or click the right mouse button to cancel the operation. After clicking the left mouse button once, the screen will update with the shifted image.

If you make a mistake select Reset to undo the shift and try again.

12.5.3. Flip

Selecting Flip from the Page menu turns the page upside down. Selecting it a second time will return the page to its original position.

12.5.4. Truncate

The size of the page (width by length, in pixels) is shown in square brackets in the bottom frame border. After a CRF has been shifted to the top of the page signature line it should not be longer than 10.5 inches. The resolution of the CRF images is 100 pixels per inch. Thus a length greater than 1050 pixels indicates that the page is longer than 10.5 inches. In such cases you should examine the bottom of the page to see if it just contains fax noise. If it does, select Truncate from the Page menu to get rid of it.

12.5.5. Trim Width

Fax machines that can accommodate pages wider than 8.5 inches require their paper guides set to 8.5 inches for proper scanning of letter-size pages. If the guides are set too wide, the paper can tilt or shift, resulting in skewed or abnormally wide images. When pages like this are received, they generally appear in the **DFrouter** tool because they are wider than the 864 pixel maximum. To route a wide page, you can trim the width of a page by selecting Trim Width from the Page menu. Two dashed vertical lines appear over the document spaced 864 pixels apart. Using the mouse, position the lines over the area of the page you want to preserve. Click the left mouse button and the page area outside the two lines is discarded, or click the right mouse button to cancel the operation. In the example shown below, the page has a horizontal resolution of 880 pixels. After trimming, the page will be the correct 864 pixels wide, and can then be routed to the appropriate study.

Figure 12.5. Trimming instructions appear at the bottom of the screen when Trim Width is selected from the Page menu.

DFrouter [valid1]

File Page Action Help

DataFax #254 Plate #001 Sequence#000

Patient Number Patient Initials F M L

BLOOD PRESSURE SCREENING VISITS

Age years Sex ☐ Male ☐ Female Race ☐ Caucasian ☐ African American ☐ Asian ☐ Other, specify: _____

Screen 1: Date day month year Reading 1: / mmHg systolic diastolic

Screen 2: Date day month year Reading 1: / mmHg systolic diastolic

Eligibility Criteria

	Yes	No
1. Age 16-79 years old.	<input type="checkbox"/>	<input type="checkbox"/>
2. Male or if female, not pregnant and using a reliable contraceptive method.	<input type="checkbox"/>	<input type="checkbox"/>

Router Pages

CRF ID

- 0441.0001001
- 0441.0005001
- 0506.0001001
- 0507.0002001
- 0507.0003001
- 0508.0001001
- 0508.0002001
- 0518.0009001
- 0520.0001001**
- 9807.0002001
- 9807.0002002
- 9807.0003001
- 9807.0003002
- 9807.0004001
- 9807.0004002
- 9807.0005001
- 9807.0005002
- 9807.0006001
- 9807.0006002
- 9807.0007001

Click left mouse button at cut point, right button to Cancel

20 routable pages (0 busy) 0520.0001001 [880x1034]

If you make a mistake, select Page > Reset to undo the Trim Width operation on the page.

12.5.6. Rotate 90° CW/CCW

You can rotate a page 90° clockwise by selecting Rotate 90° CW, or counter-clockwise by selecting Rotate 90° CCW, from the Page menu. Each time you select this menu option, the current page is rotated 90° in the specified direction. This is useful when a sender faxes a page to you sideways and you want to put it into a portrait orientation. If you make a mistake, select Page > Reset to undo all operations on the page, or rotate the page in the opposite direction to return it to its previous orientation.

12.5.7. Rotate 90° CW/CCW and scale

It is possible to process landscape pages by rotating them into portrait orientation and scaling them to fit the width of a portrait page. This is achieved using Rotate 90° CW and scale or Rotate 90° CCW and scale. Select the appropriate menu item to apply the needed rotation - the page will be rotated and then scaled automatically to fit the width of a portrait US letter page.

12.5.8. Cut

Occasionally you will receive a faxed page which is really 2 pages (or more) joined together. This will be evident by a page length count far greater than 1050 pixels, and also by the vertical scroll bar which will be reduced to half the screen length or less, depending on the number of pages joined together. In such cases you need to cut the one long page into individual pages at the appropriate page boundaries.

Selecting Cut from the Page menu brings up a small cursor you can use to position a horizontal cut. Move the mouse to the correct position and then click the left mouse button to perform the operation, or the click the right mouse button to cancel.

After the cut point is identified a second dialog box will appear, asking you to confirm this action. Click OK to cut the page or Cancel to abort. If you abort, select Reset from the Page menu to restore the original page image

Cutting a page creates 2 pages from the original one. You will see this confirmed by an increase of 1 in the number of faxes reported in the top border of the **DFrouter** main window. The top half of the cut page will become the current page displayed in **DFrouter** and the bottom half will become the next page.



Warning

The Cut action can not be undone.

12.5.9. Previous

This function moves you to the previous page in your **DFrouter** session.









12.5.10. Next

This function moves you to the next page in your **DFrouter** session.

12.5.11. Print

To print one or more pages from **DFrouter** select the page or pages you want to print in the record list, then select Print from the Page menu to get the standard DataFax Print dialog box shown below.

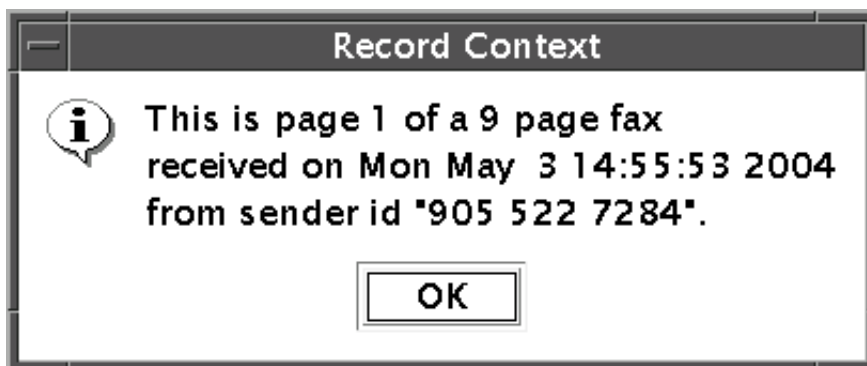
Figure 12.6. The standard DataFax Print dialog is used for printing pages from DFrouter.

Setup Printer			
Print destination			
<input checked="" type="radio"/> Print to printer:			
Printer	Host	Comment	
hp4000	telluride		
hp4100	snoqualmie		
<input type="radio"/> Print to file:			
			Browse...
Printer settings		Paper format	
<input type="radio"/> Print in color if available <input checked="" type="radio"/> Print in grayscale		Portrait  Letter (8.5x11 inches, 216x279 mm) 	
Options			
<input type="radio"/> Print all <input type="radio"/> Print selection <input type="radio"/> Print range		<input checked="" type="radio"/> Print first page first <input type="radio"/> Print last page first	
From page:	1  	Number of copies:	1  
To page:	1  		
OK		Cancel	

12.5.12. Context

This function can help you determine where an unidentified page came from. It provides the page number within the fax, the date and time on which the fax was received, and the G3 sender identification header from the fax, which is usually the sender's fax number.

Figure 12.7. The Context window.



If the `sender id` is unknown, it will be represented by "" following the sender id label. If DataFax can determine the fax number or email address of the sender's fax machine, as in the above example, it will appear following the sender id label.

12.5.13. Reset

Select this function to undo all Rotate, Rotate 90, Shift, Flip and Truncate manipulations that have been made to a page. Reset can be performed at any time as long as the page has not been dispatched from the record list, i.e. you can reset a page that you have returned to after moving on to other pages in your session of **DFrouter**. Remember that Cut is permanent and can not be undone.

12.6. The Action Menu

This menu contains all of the functions which dispatch and remove pages from **DFrouter**. All actions are immediate. Once committed they can not be undone.

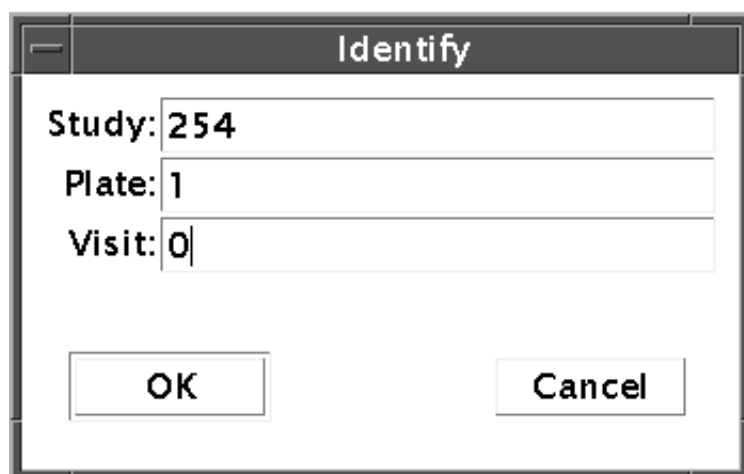
12.6.1. Rotate/Shift/Identify CRF

Most CRFs that end up in the **DFrouter** can be cleaned up, identified and routed to the appropriate study database in a single step by selecting Rotate/Shift/Identify from the Action menu. This function proceeds in 2 steps.

1. Follow the directions at the bottom of the **DFrouter** screen to identify the CRF page signature line.
2. When this has been done a second dialog box appears.

This dialog is completed with DataFax's interpretation of the bar code for Study, Plate and Visit number as shown below.

Figure 12.8. The Study, Plate and Visit confirmation dialog.

A screenshot of a dialog box titled "Identify". It contains three text input fields: "Study:" with the value "254", "Plate:" with the value "1", and "Visit:" with the value "0". At the bottom of the dialog are two buttons: "OK" and "Cancel".

Identify	
Study:	254
Plate:	1
Visit:	0
<div>OK Cancel</div>	

If the values are incorrect, enter the correct ones. If the visit number is part of the barcode, it must be entered in this dialog. If you forget to do so, a message will appear indicating that the visit number is needed and you will not be able to complete the CRF page identification until a visit number is supplied. If the visit number is the first data field on the page, it should not be entered in this dialog, and if a visit number is entered, it will be silently ignored.

To abort the Rotate/Shift/Identify action select Cancel. If OK is selected, the action is committed and cannot be undone. The page will then be sent for ICR processing and routed to the designated study database. There will be a brief pause while this is being done. Then the page will disappear, and the next page from the current **DFrouter** session will be displayed (if there is one).

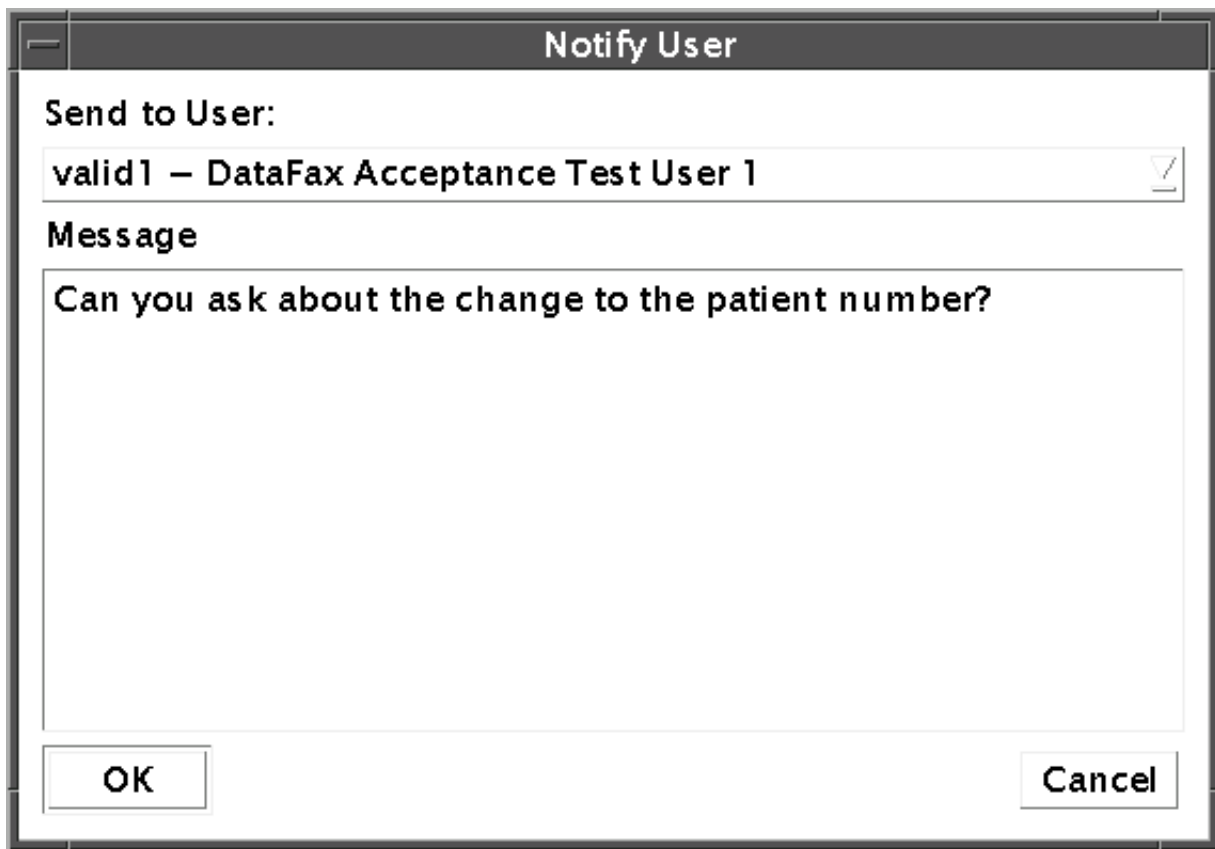
12.6.2. Identify CRF

Selecting Identify CRF allows the user assign values to the key fields of Study, Plate, and Visit without first executing the Rotate and Shift operations. When the DFrouter-Identify dialog is opened, all fields will be blank and will require user input.

12.6.3. Notify User

Select Notify User from the Action menu to send a page from **DFrouter** to another user. This is accomplished by completing the following dialog box.

Figure 12.9. The Notify User dialog box is completed in order to send a page to another user.

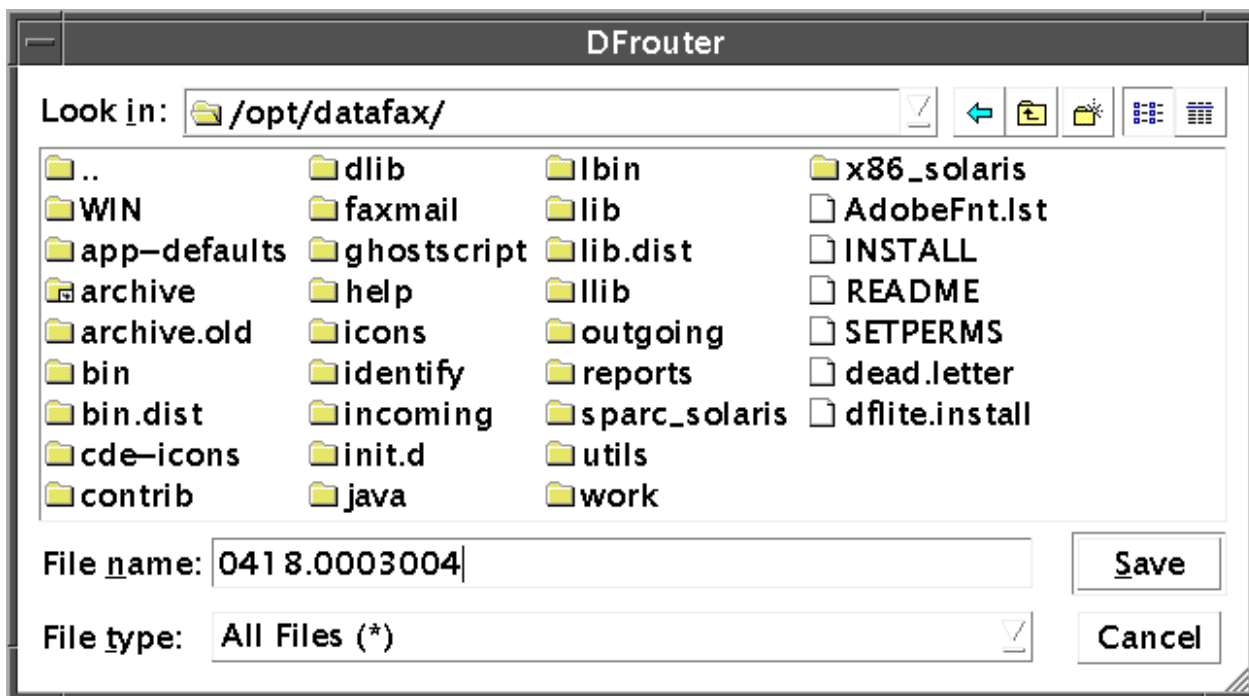


The user's login name can be selected from the scrolling list of users. A message can also be entered, but is optional. Clicking OK will deliver the message to the user by email and the page will be sent to the user's faxmail folder. The user can then use **DFfaxview** (by selecting it from the DataFax Tool Box) to see the page, and print or delete it. **DFfaxview** also allows the user to return the page to **DFrouter**. This is useful in the event that the page was misdirected, or if the user wants to process it in **DFrouter** (e.g. if it is a study CRF that needs to be routed to the correct study database).

12.6.4. Move

Select Move from the Action menu to move a page to a specified file name somewhere else in the computer file system. To move a page enter the full path name of the file you wish to save it to in the field labeled Selection at the bottom of the following dialog box.

Figure 12.10. The Move dialog allows a user to move a page from DFrouter to another location in the file system.

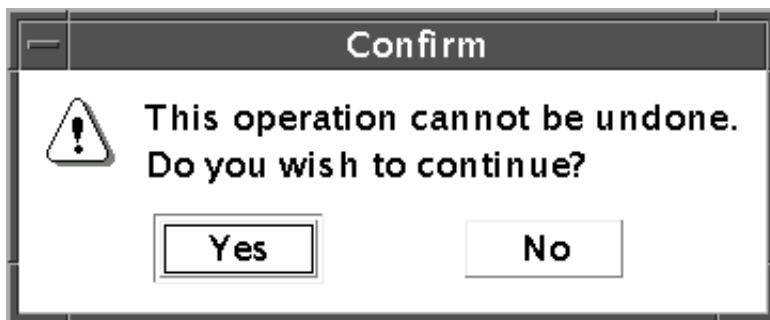


To commit the move click Save. The page will be dispatched immediately, and will disappear from **DFrouter**. To abort this operation select Cancel.

12.6.5. Discard

To discard an image, select the image to discard in the record list, then select Discard. You can discard multiple pages by selecting more than one page from the record list before selecting Discard. Discard is forever! When this action is confirmed by selecting Yes from the Discard dialog box (shown below) the page is removed from **DFrouter**, and the next page in the current **DFrouter** session will be displayed (if there is one).

Figure 12.11. A warning message appears when Discard is selected from the Action menu.

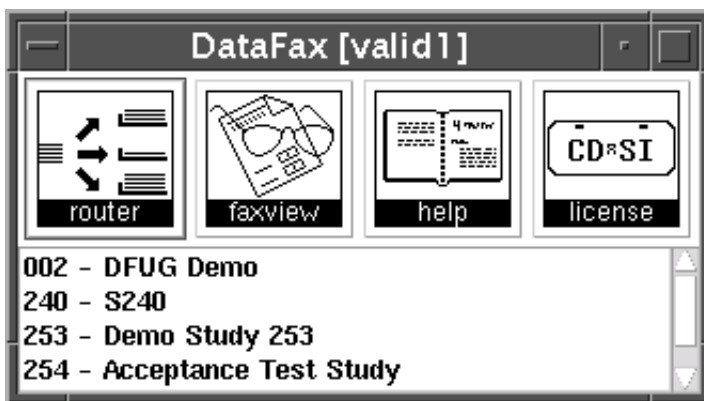


Chapter 13. Personal Faxmail Viewer

13.1. Introduction

Pages that end up in the dead letter bin can be routed to a personal fax folder for a particular user by anyone with access to the DataFax **DFrouter**. When a fax is routed to a user an email message is sent to inform the user that the fax has arrived. The person who routes the fax can include additional information, perhaps to tell the user what it's about and whether it is urgent. If you have received such an email message select the faxview icon from the DataFax Tool Box (second icon from the left in the example below) to start **DFfaxview**.

Figure 13.1. DFfaxview can be started from the DataFax User Tool Box.



13.2. DFfaxview Functions

Through **DFfaxview** you will be able to:

1. Review each fax page that is in your fax folder on screen.
2. Discard a page after reading it.
3. Print a paper copy of a page or range of pages.
4. Send the fax back to the DataFax dead letter bin for further processing with the DataFax **DFrouter**.

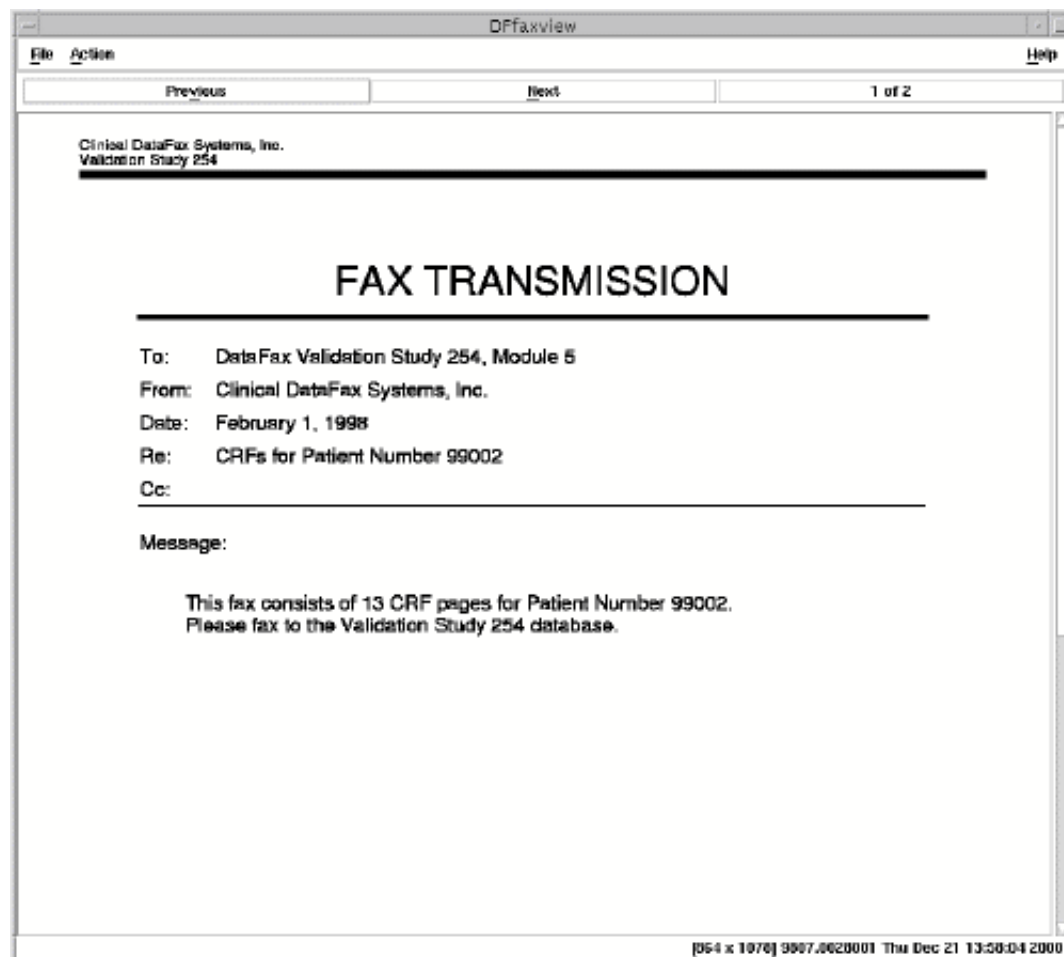
13.3. DFfaxview Main Window

When **DFfaxview** comes up it loads the first page from the user's personal fax folder into **DFfaxview**. **DFfaxview** includes 2 menus at the top of the screen. The File menu includes: Load In Box, Print and Exit, and the Action menu includes Discard and Return to Router.

Just below the File and Action menus are 2 buttons which can be used to turn to the next or previous page in the user's personal fax folder, and a third button at top right which shows the number of pages in the user's fax folder and which page is currently on display. This button can be selected to jump to a particular page.

The name assigned to the page by DataFax and the date and time that the fax arrived are shown in the lower right corner of the screen. These features are illustrated in the example shown below.

Figure 13.2. The main DFfaxview window.



13.4. The File Menu

13.4.1. Load In Box

DFfaxview loads all pages in your personal fax folder when it is opened. However, if additional pages arrive after **DFfaxview** is opened, you will need to select Load In Box from the File menu to bring them into **DFfaxview**.

13.4.2. Print

Select Print from the File menu to get the standard DataFax Print dialogue box. Using this dialog you be able to print one or more copies of the current page, or a range of pages, from your faxmail folder to the default DataFax printer or to a specified alternative destination.

13.4.3. Exit

Select Exit from the File menu to quit **DFaxview**. Any pages which have not been discarded or returned to **DFrouter** will remain in your personal fax folder and will be automatically loaded again the next time the tool is opened.

13.5. The Action Menu

13.5.1. Discard

Discard is forever! The current page is removed from your faxmail folder, and is deleted from the computer file system.

13.5.2. Send To Router

Selecting this action sends the current page back to the dead letter bin. Use this function if the fax needs to be sent to some other user or if a CRF page ends up in your faxmail folder that you wish to return it to the study CRF input stream. After selecting this action you will need to open **DFrouter** to perform either of these two actions.

Chapter 14. Customizing appearance and behavior

14.1. Introduction

This chapter is a reference for the `app-defaults` directory that is part of all standard DataFax installations. Application defaults (often named `app-defaults`) are the mechanism for customizing the appearance and behavior of DataFax tools with graphical user interfaces. These tools are:

- **DFbarcodetool**
- **DFfaxview**
- **DFidl**
- **DFqc**
- **DFreports**
- **DFsetup**
- **DFstatus**
- **DFtool**
- **DFvalidate**
- **DFviewer**
- **DataFax**

The final section of this chapter is a partial reference for some of the more commonly referenced or changed `app-defaults`. For a complete list, consult the `app-defaults` files themselves (the contents is plainly readable text) that can be found in `$DATAFAX_DIR/app-defaults/`.

14.2. Background

This level of customization is provided by the X window system, which provides a mechanism for storing default values for program resources. Resources are specified as strings that are read in from various places when an application is started.

Program components are named in a hierarchical fashion with each node in the hierarchy identified by a class and an instance name. At the top level is the class and instance name of the application itself. By convention, the class name is the same as the program name, but with the first letter capitalized (which is already true of DataFax program names). The naming of nodes below the class name is application-dependent and requires prior knowledge of the application's structure. Some of the more common elements of that structure are exposed in Section 14.3.4.

The following are examples of program resource specifications:

```
DFvalidate*validationValue.value: 1~7 (1)
DFidl.lidlWindowShell.title: Lost Data Log Tool (2)
DFviewer.filemgrWindowShell.geometry: 891x900+0+0 (3)
```

- (1) Set the range of validation levels in the validation tool to 1~7.
- (2) Set the title for the **DFidl** tool to `Lost Data Log Tool`.

- (3) Position the **DFviewer** tool at screen location 0,0 (top-left corner), set the width to 891 pixels, and set the height to 900 pixels.

14.2.1. Naming Resources

The syntax for each resource definition is:

```
ResourceName: value
```

where *ResourceName* is constructed from class names separated by one or more binding characters, . (period), * (asterisk), or ? (question mark). '.' indicates a tight binding which separates immediately adjacent components. '*' indicates a loose binding and represents any number of intervening components, including none. '?' is used to match any single component name or class. Bindings can also be thought of as pattern matching constructs. In a tight binding, such as this:

```
XTerm.title: my xterm
```

there may be no intervening components between *XTerm* and *title*, otherwise the pattern will not be matched. In a loose binding, such as this:

```
XCalc*bevel.background: black
```

any number of components may intervene between *XCalc* and *bevel* and the pattern will still be matched. A loose binding cannot end a resource specification: the final component must be specified tightly. A resource database never contains more than one entry for a given *ResourceName*. However, if there are multiple entries with the same *ResourceName*, the last one in the file is used.

When an application looks for the value of a resource, it specifies a complete path in the hierarchy, with both class and instance names. Conversely, resource values are usually defined with only partially specified names and classes using bindings. The lookup algorithm searches the resource database for the entry that most closely matches (is most specific for) the full name and class being queried. When more than one entry matches the full name and class, precedence rules are used to select exactly one. The full name and class are scanned from left-to-right (from highest level in the hierarchy to lowest), one component at a time. At each level, the corresponding component and/or binding of each matching entry is determined, and these matching components and bindings are compared according to precedence rules. Each of the rules is applied at each level, before moving to the next level, until a rule selects a single entry over all others. The rules (in order of precedence) are:

1. An entry that contains a matching component (whether name, class, or ?) takes precedence over entries that match the level in a loose binding.
2. An entry with a matching name takes precedence over both entries with a matching class and entries that match using ?.
3. An entry with a matching class takes precedence over entries that match using ?.
4. An entry preceded by a tight binding takes precedence over entries preceded by a loose binding.

14.2.2. Locating resources

Resources are obtained from the following locations (in this order):

1. **RESOURCE_MANAGER root window property.** any global resources that should be available to clients on all machines should be stored in the `RESOURCE_MANAGER` property on the root window of the first screen. This is usually taken care of when the user starts up X or a display manager.
2. **SCREEN_RESOURCES root window property.** any resources specific to a given screen (e.g. colors) that should be available to all clients on all machines should be stored in the `SCREEN_RESOURCES` property on the root window of that screen.
3. **application-specific files.** Directories named by the environment variable `XUSERFILESEARCHPATH` or the environment variable `XAPPLRESDIR`, plus directories in a standard place (usually under `/usr/openwin/lib/X11/`, but this can be overridden with the `XFILESEARCHPATH` environment variable) are searched for application-specific resources. For example, application default resources for distributed X applications are usually kept in `/usr/openwin/lib/X11/app-defaults/`. Application-specific defaults for DataFax applications are kept in `$DATAFAX_DIR/app-defaults/`.
4. **XENVIRONMENT.** Any user- and machine-specific resources may be specified by setting the `XENVIRONMENT` environment variable to the name of a resource file to be loaded by all applications. If this variable is not defined, a file named `$HOME/.Xdefaults-hostname` is looked for instead, where *hostname* is the name of the host where the application is executing. If a `$HOME/.Xdefaults-hostname` file cannot be located, then a `$HOME/.Xdefaults` file is used, if present.



Important

It is important to note that the application-specific defaults for an application are loaded and parsed at the time that the application is started. This allows an application to inherit the most current defaults each time that it starts. This is however not true of user-specific defaults. User-specific defaults are loaded and parsed at the time that a user connects to an X display server. Subsequent changes to a `.Xdefaults` or `.Xdefaults-hostname` file will not take effect until the user exits and re-connects to the display server. This behavior can be superseded by the use of the **xrdb** program, however a description of this program is outside of the scope of this manual.

14.3. DataFax defaults

DataFax application defaults can be specified at three levels: application level, study level, and user-specific. The application level is the highest and the user level is the lowest. Any default can be overridden at a lower level by a different setting for the same default already specified at a higher level. For example, setting a user-specific default will override the same default specified at the application level.

14.3.1. Application-level defaults

Application-level defaults are stored in multiple files in the `$DATAFAX_DIR/app-defaults` directory, each file corresponding to a different application. These files are provided with the software and typically should not be changed or removed, but rather overridden at one of the other two levels.

The filename provides the class name and so it is not necessary to repeat the class name in the resource specification.

14.3.2. Study-level defaults

Study-level defaults are stored in multiple files in the `$STUDY_DIR/app-defaults` directory, again each file corresponding to a different application, and where `$STUDY_DIR` corresponds to a specific study's directory. Resources specified in these files override the application-level and set the resource value for a single study.

For example, the application-level default for QC status is set in `$DATAFAX_DIR/app-defaults/DFvalidate` by the resource string:

```
*.defaultQcIsExternal: True
```

This resource specifies that QC notes should be created by default with their status set to external (as opposed to internal). An individual study may override this setting, creating QC notes by default with their status set to internal, with the resource string:

```
*.defaultQcIsExternal: False
```

specified in `$STUDY_DIR/app-defaults/DFvalidate`.

It is not necessary, or recommended, to repeat all of the resource settings from the application level, only those that are being changed. Also, as is true at the application level, the filename provides the class name and so it is not necessary to repeat the class name in the resource specification.

14.3.3. User-level defaults



User-level defaults are specified in the single file `~/.Xdefaults`, stored in the user's home directory. Since all of the resources are specified in a single file it is important to include the class name in the resource specification. For example, for an individual user to set the QC status to external by default, the resource string:

```
DFvalidate*.defaultQcIsExternal: True
```

would be specified in `~/.Xdefaults`. Remember also in this case that the changed resource setting is not reflected in the user environment until the user exits and re-connects with the display server.

14.3.4. Defaults by application

Table 14.1. DFbarcodetool

Name	Value	Description
*.bartoolStudyLabelTextField.value	DataFax #	<p>This resource controls the descriptive text that appears as the entry for Study Label in the DFbarcodetool dialog. This descriptive label precedes the DataFax study number, and the two combined are printed below the study number component of a CRF barcode when the component is generated. DataFax # is the default descriptive text. The value DataFax # can be edited within the app-defaults resource to create a customized text label. The text of the label may also be edited in the DFbarcodetool dialog itself and applied when the barcode is created. Edits made in the DFbarcodetool dialog will override any default label defined in app-defaults.</p> <p> Note</p> <p>The numeric value of the study number component <i>always</i> appears <i>after</i> the descriptive label. If the descriptive label is left blank, only the number will be printed below the barcode.</p>
*.bartoolPlateLabelTextField.value	Plate #	<p>This resource controls the descriptive text that appears as the entry for Plate Label in the DFbarcodetool dialog. This descriptive label precedes the plate number, and the two combined are printed below the plate number component of a CRF barcode when the component is generated. Plate # is the default descriptive text. The value Plate # can be edited within the app-defaults resource to create a customized text label. The text of the label may also be edited in the DFbarcodetool dialog itself and applied when the barcode is created. Edits made in the DFbarcodetool dialog will override any default label defined in app-defaults.</p> <p> Note</p> <p>The numeric value of the plate number component <i>always</i> appears <i>after</i> the descriptive label. If the descriptive label is left blank, only the number will be printed below the barcode.</p>


Name	Value	Description
*.bartoolVisitLabelTextField.value	Visit #	<p>This resource controls the descriptive text that appears as the entry for Visit Label in the DFbarcode dialog. This descriptive label precedes the visit number, and the two combined are printed below the visit number component of a CRF barcode when the component is generated. Visit # is the default descriptive text. The value Visit # can be edited within the app-defaults resource to create a customized text label. The text of the label may also be edited in the DFbarcode dialog itself and applied when the barcode is created. Edits made in the DFbarcode dialog will override any default label defined in app-defaults.</p> <p> Note</p> <p>The numeric value of the visit number component <i>always</i> appears <i>after</i> the descriptive label. If the descriptive label is left blank, only the number will be printed below the barcode.</p>

Table 14.2. DFfaxview

Name	Value	Description
*.faxviewWindow.height	900	This resource controls the height of the main DFfaxview window, in pixels.
*.selectList.visibleItemCount	10	The number of rows to display in the visible portion of the CRF selection dialog. The list will be sized large enough to accommodate all of the rows, but only this number will be visible, by default. The user can still resize the list to make it larger.

Table 14.3. DFidl

Name	Value	Description
*.ldlViewScrolledList.visibleItemCount	3	The number of rows to display in the visible portion of the lost data record retrieval window. The list will be sized large enough to accommodate all of the rows, but only this number will be visible, by default. The user can still resize the list to make it larger.

Table 14.4. DFqc

Name	Value	Description
<code>*.qcWindowShell.height</code>	746	This resource controls the default height of the main DFqc tool window, in pixels.
<code>*.viewToggleButton1.set</code>	True	This resource controls the toggle button setting for the n^{th} toggle button appearing in the right-hand column of the View QC Notes dialog. By default, this value is set to <code>True</code> resulting in the n^{th} toggle button being selected when the View QC Notes dialog is opened. Setting the resource value to <code>False</code> results in an unchecked toggle button upon opening of the View QC notes dialog. When the dialog is subsequently updated only those attributes for which the corresponding toggle is set to <code>True</code> will be displayed.

Table 14.5. DFreports

Name	Value	Description
<code>*.reportsList.visibleItemCount</code>	8	The number of reports visible in the reports list window. All of the reports are contained in the list but only this many are visible at any time without scrolling.
<code>*reportsOutputText.autoShowCursorPosition</code>	true	Should the contents of the reports output window always be scrolled so that the cursor (which is at the end of the output) is always visible? Setting this default to <code>false</code> ensures that the scroll position of the reports output window is under the user's control, and does not scroll automatically to make the end of the output visible.

Table 14.6. DFsetup

Name	Value	Description
<code>*.setupWindowShell.height</code>	900	This resource controls the height of the main DFsetup window, in pixels.
<code>*.varWindow.geometry</code>	-0+0	When the Variable dialog appears, position it on the screen using the location -0 (meaning right-edge) and +0 (meaning top-edge).
<code>*.varScrolledWindow.width</code>	330	This resource controls the width, in pixels, of the variable definition dialog in DFsetup .
<code>*.varScrolledWindow.height</code>	800	This resource controls the height, in pixels, of the variable definition dialog in DFsetup .

Name	Value	Description
*.varLegalValue.maxLength	200	This resource sets the maximum number of characters that can be entered and stored in the Legal field of the variable definition dialog in DFsetup .
*.procPreValue.maxLength	200	This resource sets the maximum length of the On field enter edit check specification field in the variable definition dialog.
*.procPostValue.maxLength	200	This resource sets the maximum length of the On field exit edit check specification field in the variable definition dialog.
*.procBeginValue.maxLength	200	This resource sets the maximum length of the On plate enter edit check specification field in the variable definition dialog.
*.procEndValue.maxLength	200	This resource sets the maximum length of the On plate exit edit check specification field in the variable definition dialog.
*.printStudyAllToggleButton.set	True	This resource controls the toggle button setting for the option Pages: All in the Print dialog of DFsetup . The Print dialog is opened by selecting Print from the Study in DFsetup . By default, this value is set to True resulting in the toggle button for Pages: All being selected when the Print dialog is opened. With the Pages: All toggle button selected, all pages in DFsetup will be printed. Setting the resource value to False results in the toggle button being de-selected upon opening of the Print dialog.
*.printStudy1UpToggleButton.set	True	This resource controls the toggle button setting for the option Format: 1 Up in the Print dialog of DFsetup . The Print dialog is opened by selecting Print from the Study in DFsetup . By default, this value is set to True resulting in the toggle button for Format: 1 Up being selected when the Print dialog is opened. With the Format: 1 Up toggle button selected, pages will be printed with one image per page. Setting the resource value to False results in the toggle button being de-selected upon opening of the Print dialog.

Name	Value	Description
<code>*.printStudyToFileButton.set</code>	False	This resource controls the toggle button setting for the option Print Only To File in the Print dialog of DFsetup . The Print dialog is opened by selecting Print from the Study in DFsetup . By default, this value is set to <code>False</code> resulting in the toggle button for Print Only To File being de-selected when the Print dialog is opened. With the Print Only To File toggle button de-selected, pages will be printed to the default printer rather than to the specified filename. The default printer is set to the printer name located in the Printer Name field of the Print dialog. Setting the resource value to <code>True</code> results in the Print Only To File toggle button being selected upon opening of the Print dialog.
<code>*.editChecksD*editChecksText.rows</code>	24	This resource controls the number of text lines visible at one time in the Input File section of the Edit Checks dialog.
<code>*.editChecksD*editChecksText.columns</code>	80	This resource controls, by number of characters, the width of the Input File window in the Edit Checks dialog.
<code>*.editChecksD*compileOutputText.rows</code>	10	This resource controls the number of text lines visible at one time in the Output compiler window of the Edit Checks dialog.
<code>*.editChecksD*compileOutputText.columns</code>	80	This resource controls, by number of characters, the width of the Output compiler window in the Edit Checks dialog.

Table 14.7. DFstatus

Name	Value	Description
<code>*.userStatusList.visibleItemCount</code>	5	This resource controls the number of user names that are visible at any one time in the list of users present in the User Status window.

Name	Value	Description
<code>*.serverStatusToggle.set</code>	True	This resource controls the toggle button setting for the Server Status window in the DFstatus tool dialog. By default, this value is set to <code>True</code> resulting in the Server Status toggle button being selected when the DFstatus dialog is opened. If Server Status is selected, information about the DataFax server will be displayed when Update Now or Update Periodically is chosen from the Action menu. Setting the resource value to <code>False</code> results in an unchecked Server Status toggle button upon opening of DFstatus . Server status information will hence not be updated when updating the status of the database.
<code>*.userStatusToggle.set</code>	True	This resource controls the toggle button setting for the User Status window in the DFstatus tool dialog. By default, this value is set to <code>True</code> resulting in the User Status toggle button being selected when the DFstatus dialog is opened. If User Status is selected, information about user activity in the database will be displayed when Update Now or Update Periodically is chosen from the Action menu. Setting the resource value to <code>False</code> results in an unchecked User Status toggle button upon opening of DFstatus . User status information will hence not be updated when updating the status of the database.
<code>*.databaseStatusToggle.set</code>	False	<p>This resource controls the toggle button setting for the Database Status window in the DFstatus tool dialog. By default, this value is set to <code>True</code> resulting in the Database Status toggle button being selected when the DFstatus dialog is opened. If Database Status is selected, information about the contents of the database will be displayed when Update Now or Update Periodically is chosen from the Action menu. Setting the resource value to <code>False</code> results in an unchecked Database Status toggle button upon opening of DFstatus. Information about the contents of the database will hence not be updated when updating the status of the database.</p> <p>Users with very large databases may want to toggle this option off so that the tool appears more quickly when started.</p>

Table 14.8. DFvalidate

Name	Value	Description
<code>*.validateWindowShell.geometry</code>	+0+0	When the main window appears, it is positioned at +0 horizontally (meaning the left-edge is at 0 pixels), and +0 vertically (meaning the top-edge is at 0 pixels).
<code>*.viewDataButton.set</code>	True	This resource controls the toggle button setting for the Data option in the Window menu. By default, this value is set to <code>True</code> resulting in the Data toggle button being enabled when the Window is selected. If enabled, the Data window (bottom half of the validation tool split screen) will be displayed. If disabled (value is set to <code>False</code>), the Data window will not be displayed.
<code>*.viewCrfButton.set</code>	True	This resource controls the toggle button setting for the CRF option in the Window menu. By default, this value is set to <code>True</code> resulting in the CRF toggle button being enabled when the Window is selected. If enabled, the CRF window (top half of the validation tool split screen) will be displayed. If disabled (value is set to <code>False</code>), the CRF window will not be displayed.
<code>*.viewPageButton.set</code>	True	This resource controls the toggle button setting for the Data/CRF Separator option in the Window menu. By default, this value is set to <code>True</code> resulting in the Data/CRF Separator toggle button being enabled when the Window is selected. If enabled, the Data/CRF Separator window will be displayed. If disabled (value is set to <code>False</code>), the Data/CRF Separator window will not be displayed.
<code>*.statusButton.set</code>	True	This resource controls the toggle button setting for the Record option in the Window menu. By default, this value is set to <code>True</code> resulting in the Record toggle button being enabled when the Window is selected. If enabled, the Record List window (to the right of the main DFvalidate window) will be displayed. If disabled (value is set to <code>False</code>), the Record List window will not be displayed.

Name	Value	Description
<code>*.viewMetaButton.set</code>	True	This resource controls the toggle button setting for the Meta option in the Window menu. By default, this value is set to <code>True</code> resulting in the Meta toggle button being enabled when the Window is selected. If enabled, the Meta window (located to the bottom right of the main DFvalidate window), will be displayed. If disabled (value is set to <code>False</code>), the Meta window will not be displayed.
<code>*.viewVarButton.set</code>	False	This resource controls the toggle button setting for the Variable option in the Window menu. By default, this value is set to <code>False</code> resulting in the Variable toggle button being disabled when the Window is selected. If disabled, the Variable window (located to the bottom left of the main DFvalidate window), will not be displayed when DFvalidate is started. If enabled (value is set to <code>True</code>), the Variable window will be displayed upon starting DFvalidate .
<code>*.viewClipButton.set</code>	False	This resource controls the toggle button setting for the Clipboard option in the Window menu. By default, this value is set to <code>False</code> resulting in the Clipboard toggle button being disabled when the Window is selected. If disabled, the Clipboard window (located at the top of the main DFvalidate window) will not be displayed when DFvalidate is started. If enabled (value is set to <code>True</code>), the Clipboard window will be displayed upon starting DFvalidate .
<code>*.viewEditsButton.set</code>	False	This resource controls the toggle button setting for the Edit Checks Log option in the Window menu. By default, this value is set to <code>False</code> resulting in the Edit Checks Log toggle button being disabled when the Window is selected. If disabled, the Edit Checks Log window (located to the upper right of the main DFvalidate window), will not be displayed when DFvalidate is started. If enabled (value is set to <code>True</code>), the Edit Checks Log window will be displayed upon starting DFvalidate .
<code>*.crfScrolledWindow.height</code>	395	The height of the visible portion of the faxed CRF window. To maximize the amount of screen space used for a particular monitor's resolution, adjust this value to be the result of the calculation $(vertical_resolution - 110) / 2$.

Name	Value	Description
<code>*.dataScrolledWindow.height</code>	395	The height of the visible portion of the data entry window. To maximize the amount of screen space used for a particular monitor's resolution, adjust this value to be the result of the calculation (<i>vertical_resolution</i> - 110) / 2.
<code>*.statusRecordsList.visibleItemCount</code>	55	This resource controls the number of record entries visible at one time in the Record List window.
<code>*.viewMetaWindow.geometry</code>	-0-0	Position the View QC dialog at position -0 (right-edge) and -0 (bottom-edge) when it is displayed.
<code>*.statusValue*clean.set</code>	True	<p>This resource controls the toggle button setting for the Status clean option in the Build dialog when record retrieval is to be performed by the By Data Fields setting. By default, this value is set to <code>True</code> resulting in the Status clean toggle button being selected when retrieval By Data Fields is selected and the Build dialog is opened. This setting will allow the retrieval of all primary, clean records in the database, that match on any other specified selection criteria. Setting the resource value to <code>False</code> results in an unchecked Status clean toggle button upon opening of the Build dialog, and subsequent exclusion of primary, clean records from record retrievals.</p> <p>There are similar toggle for dirty and error primary record statuses, as well as clean, dirty, and error secondary record statuses.</p>
<code>*.prefAutoClearClipboard.set</code>	True	This resource controls the toggle button setting for the Reset clipboard after each variable option in the Preferences dialog. By default, this value is set to <code>True</code> resulting in the toggle button being enabled when the Preferences dialog is opened. Setting the resource value to <code>False</code> results in the toggle button being disabled upon opening of the Preferences dialog. A description of the Reset clipboard after each variable preference setting can be found in Preferences.

Name	Value	Description
<code>*.prefAutoTab.set</code>	True	This resource controls the toggle button setting for the Advance to next field after keying complete value option in the Preferences dialog. By default, this value is set to <code>True</code> resulting in the toggle button being enabled when the Preferences dialog is opened. Setting the resource value to <code>False</code> results in the toggle button being disabled upon opening of the Preferences dialog. A description of the Advance to next field after keying complete value preference setting can be found in Preferences.
<code>*.prefAutoResolveMissingQC.set</code>	False	This resource controls the toggle button setting for the Resolve missing value QCs as data is corrected option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Resolve missing value QCs as data is corrected preference setting can be found in Preferences.
<code>*.prefAutoResolveIllegalQC.set</code>	False	This resource controls the toggle button setting for the Resolve illegal value QCs as data is corrected option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Resolve illegal value QCs as data is corrected preference setting can be found in Preferences.
<code>*.prefAutoExpandText.set</code>	True	This resource controls the toggle button setting for the Expand text variables to display maximum stored text option in the Preferences dialog. By default, this value is set to <code>True</code> resulting in the toggle button being enabled when the Preferences dialog is opened. Setting the resource value to <code>False</code> results in the toggle button being disabled upon opening of the Preferences dialog. A description of the Expand text variables to display maximum stored text preference setting can be found in Preferences.

Name	Value	Description
<code>*.prefConfirmSignOffChanges.set</code>	True	This resource controls the toggle button setting for the Ask for confirmation before saving record changes option in the Preferences dialog. By default, this value is set to <code>True</code> resulting in the toggle button being enabled when the Preferences dialog is opened. Setting the resource value to <code>False</code> results in the toggle button being disabled upon opening of the Preferences dialog. A description of the Ask for confirmation before saving record changes preference setting can be found in Preferences.
<code>*.prefConfirmSwitchChanges.set</code>	False	This resource controls the toggle button setting for the Ask for confirmation to switch record without saving changes option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Ask for confirmation to switch record without saving changes preference setting can be found in Preferences.
<code>*.prefConfirmLevel.set</code>	False	This resource controls the toggle button setting for the Ask for confirmation of the sign-off validation level option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Ask for confirmation of the sign-off validation level preference setting can be found in Preferences.
<code>*.prefDisplayMissingCrfMessage.set</code>	False	This resource controls the toggle button setting for the Display a warning dialog for each record with a missing CRF option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Display a warning dialog for each record with a missing CRF preference setting can be found in Preferences.

Name	Value	Description
<code>*.prefEditChecks.set</code>	True	This resource controls the toggle button setting for the Execute all defined edit checks option in the Preferences dialog. By default, this value is set to <code>True</code> resulting in the toggle button being enabled when the Preferences dialog is opened. Setting the resource value to <code>False</code> results in the toggle button being disabled upon opening of the Preferences dialog. A description of the Execute all defined edit checks preference setting can be found in Preferences.
<code>*.prefQcSort.set</code>	False	This resource controls the toggle button setting for the Sort records within sets by QC note sort order option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Sort records within sets by QC note sort order preference setting can be found in Preferences.
<code>*.prefDdeSort.set</code>	False	This resource controls the toggle button setting for the Sort records within DDE batch files chronologically option in the Preferences dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the Preferences dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the Preferences dialog. A description of the Sort records within DDE batch files chronologically preference setting can be found in Preferences.
<code>*.defaultQcIsExternal</code>	True	This resource controls the setting for the Use option in the QC Add dialog. By default, the value of the Use button is set to External as a result of this resource evaluating to <code>True</code> . Hence, each time a QC note is added to a data field, it is marked for external use, by default. Setting the resource value to <code>False</code> results in the Use option being set to Internal for newly added QC notes.

Name	Value	Description
<code>*.defaultQcIsRefax</code>	True	This resource controls the setting for the Type option in the QC Add dialog. By default, the value of the Type button is set to Refax as a result of this resource evaluating to <code>True</code> . Hence, each time a QC note is added to a data field, it's type is Refax, which indicates that this query will be included in the Refax List of a DataFax Quality Control report. Setting the resource value to <code>False</code> results in the Type option being set to Q/A for newly added QC notes. All such QC notes will be included in the Question and Answer section of a DataFax Quality Control report.

Table 14.9. DFviewer

Name	Value	Description
<code>*.filemgrWindow.height</code>	900	This resource controls the height of the main DFviewer window, in pixels.
<code>*.viewQcButton.set</code>	False	This resource controls the toggle button setting for the QCs option in the Window menu. By default, this value is set to <code>False</code> resulting in the QCs option being disabled when the Window is selected. If disabled, the QCs window will not be displayed when the DFviewer tool is started. If enabled, (value is set to <code>True</code>), the QCs window will be displayed.
<code>*.viewByD*prefQcSort.set</code>	False	This resource controls the toggle button setting for the Sort records within sets by QC note sort order option in the View CRFs By dialog. By default, this value is set to <code>False</code> resulting in the toggle button being disabled when the View CRFs By dialog is opened. Setting the resource value to <code>True</code> results in the toggle button being enabled upon opening of the View CRFs By dialog. This setting has the same functionality as the Sort records within sets by QC note sort order toggle button displayed in the Preferences dialog in DFvalidate . A description of the Sort records within sets by QC note sort order preference setting can be found in Preferences.