

# Standard Reports Guide

## Standard Reports Guide

Release 5.3.0

Publication date Jul 09, 2020

Copyright © 2020 DF/Net Research, Inc.

### Abstract

This guide describes the standard DFdiscover reports that are distributed with DFdiscover. Synopses and detailed descriptions are provided for each report.

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 DF/Net Research, Inc. Permission is granted for internal re-distribution of this publication by the license holder and their 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. DF/Net Research, 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.

Google Play and the Google Play logo are trademarks of Google LLC. Android is a trademark of Google LLC.

App Store is a trademark of Apple Inc.

---

# Table of Contents

Preface .....	ix
1. Getting Help .....	ix
2. Conventions .....	ix
1. Introduction .....	1
1.1. Restrictions .....	1
2. Reference Pages .....	2
2.1. Organization of Reference Pages .....	2
2.1.1. Name .....	2
2.1.2. Synopsis .....	2
2.1.3. Description .....	3
2.1.4. Options .....	3
2.1.5. Examples .....	3
2.1.6. See Also .....	3
2.1.7. Limitations .....	3
2.2. Alphabetical Listing, DFdiscover Reports .....	3
2.3. Alphabetical Listing, Legacy Reports .....	23
A. Copyrights - Acknowledgments .....	152
A.1. External Software Copyrights .....	152
A.1.1. DCMTK software package .....	152
A.1.2. Jansson License .....	152
A.1.3. Mimecode .....	153
A.1.4. RSA Data Security, Inc., MD5 message-digest algorithm .....	153
A.1.5. mpack/munpack .....	153
A.1.6. TIFF .....	153
A.1.7. PostgreSQL .....	154
A.1.8. OpenSSL License .....	154
A.1.9. Original SSLeay License .....	155
A.1.10. gawk .....	155
A.1.11. Ghostscript .....	159
A.1.12. MariaDB and FreeTDS .....	159
A.1.13. QtAV .....	165
A.1.14. FFmpeg .....	165
A.1.15. c3.js .....	165
A.1.16. d3.js .....	166
Index .....	167

## List of Figures

2.1. Legend of suffix codes and their meaning .....	59
2.2. Alternative Report from -a .....	69

## List of Tables

1.1. Conformance with Access Restrictions for Standard Reports .....	1
2.1. Notation for report options .....	2
2.2. Variables available for use in QCcovers, QCmessages, and QCtitles .....	94

## List of Examples

2.1. Report the audit trail for CRF page 1525/0016001 .....	25
2.2. Show all documents received on Jan 16, 2018 .....	27
2.3. Show all journal entries for the pages in document 9803/0027 .....	28
2.4. Show new data records, modifications to existing records, and all internal and external queries added or modified, between Dec 1,2017 and today's date .....	32
2.5. Show modifications made to existing external queries by jane between Dec 1 and Dec 24, 2015 .....	32
2.6. Show all modifications, excluding queries, made after a data record reached validation level 6 .....	32
2.7. Show all modifications made to the database between 8am and 9am this morning .....	32
2.8. Show modifications made during the year 2015 to fields 15 through 18 and 22 of external queries .....	32
2.9. Show modifications made at validation levels 4 through 7, between July 1, 2015 and today's date .....	32
2.10. Show modifications made after validation level 6, by jane, between Jan 1, 2015 and today's date .....	33
2.11. report on sites 1-20 for visits 0-2,4 labeling each site with the first 4 characters of the site name (which is in field 3 of the site record) .....	40
2.12. Check key fields and initials (found in field 8) on plates 1-3 .....	46
2.13. Check key fields and initials on final and missed records for plates 1-10 .....	47
2.14. Check initials but not the keys on all primary records of plates 11-20 .....	47
2.15. Check key fields on plates 1-2 using only the study schema as the source for legal subject IDs. ....	47
2.16. Check only primary records on plates 1-5 .....	51
2.17. Check queries and split any inconsistent records .....	51
2.18. Report output for an incorrectly modified version of study 245 .....	57
2.19. List CRF information for all subjects at site 99 .....	59
2.20. List, with default formatting, fields 9 and 21-25 from plate 1, and fields 10 and 12 from plate 5 .....	62
2.21. From plate 1, list fields 8 and 9, right justified in columns that are 10 and 12 characters wide, respectively .....	62
2.22. Normalize the 3 blood pressure measurements in fields 21 through 26 on plate 1 .....	62
2.23. List cases in which the 5 <sup>th</sup> field extracted equals 1, where $x[i]$ is the $i^{\text{th}}$ field in the next -x, -X or -R .....	63
2.24. List field 8 through 10 of plate 1 for subjects 99000-99999 .....	63
2.25. Print a page break and column titles every 50 lines .....	63
2.26. Print every subject on a new page .....	63
2.27. List fields 9 to 12 from plate 1, and fields 15, 12 and 16 (in that order) from plate 2 .....	64
2.28. List fields 8 and 9 from plate 1, formatted using field widths of 5 and 12 characters respectively .....	64
2.29. List fields 12 to 16 from plate 5 where field 12 of plate 5 equals 1, and fields 21 and 22 from plate 8 .....	64
2.30. List multiple events, e.g. adverse events, for each subject .....	65
2.31. Using -e to end previous selection criteria .....	65
2.32. Display overdue visits and missing plates for site 1 .....	66
2.33. Display overdue visits and missing plates, by their name, for site 1 .....	66
2.34. First Tabular Report .....	68
2.35. Second Tabular Report .....	69
2.36. show visit scheduling info for subjects still under follow-up at sites 10 and 99 .....	70
2.37. Select subjects from sites 40-44 for whom any visit was performed early or late by 10+ days, or with a visit currently overdue by 30+ days, or with a visit overdue by an unknown duration .....	78
2.38. Show visits 101-199 for subjects who have completed all follow-up .....	78
2.39. Show baseline, scheduled follow-up and termination visits only .....	78
2.40. Show cycles that are not expected but which nonetheless are in progress or have been completed .....	78
2.41. Show all info tags including study day .....	78
2.42. Send two unsent Query Reports, created for sites 120 and 122 on Aug 15/04 .....	82
2.43. Fax an internal report named test to 555-1212 from dan at 2am .....	82
2.44. Send an unsent Query Report created for site 123 on June 3/04, retrying once, if necessary, after a delay of 30 minutes .....	82
2.45. Email a previously sent external Query Report created for site 150 on June 25/04 to dan@hotmail.com. ....	82
2.46. Re-send all failed Query Reports created on October 3, 2017. ....	82

2.47. Email the unsent external Query Report created for site 123 on August 8, 2010 to dan@hotmail.com with a customized signature line .....	82
2.48. Display the last 5 lines from .....	85
2.49. Display the status for all Query Reports created on 17/03/23, sorted by report ID .....	85
2.50. Display the last 20 FAILED transmissions .....	85
2.51. Display any FAILED transmissions among the last 20 transmissions registered .....	85
2.52. Print specified unsent Query Reports .....	86
2.53. Print all sent Query Reports created on 99/04/15 .....	86
2.54. Print the unsent external Query Report created for Site 123 on August 8, 2010 with a customized signature line label. ....	87
2.55. The Subject Status List .....	89
2.56. The Fax/Refax List .....	89
2.57. The Question and Answer List .....	90
2.58. Titles, Cover Sheets and Messages .....	91
2.59. English and French version of a cover sheet .....	95
2.60. Typical use of QMessages .....	96
2.61. Create reports with only Fax/Refax List and Question & Answer List sections for sites 15 and 21-24 .....	99
2.62. Create reports that include unresolved queries sent >30 days ago .....	99
2.63. Create an internal subject status report .....	100
2.64. Exclude completed subjects with no outstanding queries .....	100
2.65. Create an internal subject status report using the first visit from the most recent cycle .....	100
2.66. Create an internal report including all internal queries plus external unresolved queries .....	100
2.67. Include the Fax/Refax List and Question & Answer List sections (but not the status list) and put Q&As on a new page for each subject .....	100
2.68. Include a cover sheet and the Subject Status Summary for all subjects, and put all Refax queries only on a new page for each subject .....	100
2.69. Include the Subject Status Summary for all subjects followed by a new page for each subject with unresolved queries, with both the refax and Q&A notes together on a separate page for each subject .....	100
2.70. Include the Fax/Refax List and Question & Answer List queries (but not the Subject Status Summary) together on a separate page for each subject. Sequentially number the queries within subject ID and place the problem description first for each query. ....	100
2.71. Create a report for sites 10 through 15 only, containing only the cover sheet .....	100
2.72. Include only queries which are on plates 1 and 2 with sequence number 0 and which have reached validation levels 4 through 7 .....	101
2.73. Create reports listing only the Fax/Refax List and Question & Answer List sections and that include only problem types 3, 4 and 21 .....	101
2.74. Create reports listing only refax requests on plates 1 through 13 inclusive that were created between April 1 and April 28, 2018 .....	101
2.75. Create reports for sites 11 and 23 through 30 inclusive that include only the Fax/Refax List and Question & Answer List sections .....	101
2.76. Create an internal report named <code>int</code> that contains query categories 1 through 6, refax and Q&A queries, for visit 3 and plate 11 only .....	101
2.77. Create an internal report named <code>int</code> that contains internal, unresolved queries of refax type .....	101
2.78. Create a report for site 10 that includes only the Subject Status Summary section, subjects with ongoing follow-up, the entry visit as the baseline visit, and the last follow-up visit as any visit that has a <code>Visit Date</code> .....	101
2.79. Create a standard 3-part Query Report for all subjects of Site 5 that have external, unresolved queries on field 9 of visit 0, plate 1, on all fields of visits 1-20, plate 5, and on all fields of all plates at visit 50. ....	101
2.80. Create an internal Query Report named <code>monitor_rpt</code> that includes only internal queries having either or both of the strings <code>PI</code> or <code>COORD</code> in their <code>Note</code> field. ....	102
2.81. Register 2 specified Query Reports as sent .....	107
2.82. Register as sent all of the unsent reports created on September 22, 2017 .....	107
2.83. List all unsent Query Reports .....	109
2.84. Determine if a specified report is registered as sent .....	109
2.85. List all Query Reports sent to site ID 001 .....	109

2.86. List sent reports created on Aug 15,2011 .....	109
2.87. List all internal Query Reports whose name begins with stat.91 .....	110
2.88. Show results from last execution of <b>DF_QCupdate</b> then quit .....	114
2.89. Update the Query database and all summary files .....	114
2.90. Delete all overdue visit and missing page queries .....	115
2.91. Delete missing page queries created by DF_QCupdate and dfaddmpqc .....	115
2.92. Display the specified unsent Query Report .....	117
2.93. Display the specified sent Query Report .....	117
2.94. Display the specified internal Query Report .....	117
2.95. Listing for a sites database with 3 sites .....	118
2.96. Display data dictionary information for all variables on plates 7, 8 to 10 inclusive, and 22 .....	120
2.97. Display the description of all study plates, plus all fields on plates 1, 2 and 3 .....	120
2.98. Display data dictionary information for all variable(s) with DRUG in the variable name .....	120
2.99. Select and display all variable(s) with sore throat in the variable description .....	120
2.100. List variables defined using the Visit Date attribute .....	120
2.101. Display data dictionary information for all variables with edit checks on plates 1, 2 and 3. ....	121
2.102. Select variables from plate 2 containing the string date in any attribute .....	124
2.103. List variables from plate 2 that have a variable type attribute beginning with date .....	124
2.104. List all required date variables from plates 1 and 2 .....	125
2.105. List all dates and all required variables from plates 1 and 2 .....	125
2.106. List the first 7 variables from all plates .....	125
2.107. Display the name, description, and legal values for all variables on plates 1, 3 and 6 through 8 that have the string VisitDate in an attribute .....	125
2.108. -i vmap option .....	126
2.109. -i visits option .....	127
2.110. -i vptbl option .....	128
2.111. -i pvtbl option .....	128
2.112. -i vdates option .....	129
2.113. -i etp option .....	129
2.114. -i cterm option .....	129
2.115. -i cplate option .....	130
2.116. -i cvisit option .....	130
2.117. -i ccycle option .....	130
2.118. Summarize CRFs validated by user1 during Jan 2016 .....	133
2.119. Summarize CRFs validated by user1 between Dec 1,2015 and Apr 1,2016 .....	133
2.120. Summarize CRFs validated by user1 during the first half of June 2015 .....	133
2.121. Summarize the fax traffic for the last 10 weeks .....	136
2.122. Display, in reverse chronological order, the disk usage for study weeks from the current week back through the first week of 2017. ....	137
2.123. Display, the disk usage from the 34 <sup>th</sup> week of 1999 to the 34 <sup>th</sup> week of 2017, inclusive .....	138
2.124. Summarize processing time for documents received in the first half of November 2017 .....	141
2.125. Show all queries journaled by user1 from Jan 6 to Jan 9/18 .....	142
2.126. Tabulate external queries on plates 3 through 7 and 10 .....	147
2.127. Tabulate fields which have 5 or more queries .....	147
2.128. Tabulate queries for fields with "BP" in their description .....	147
2.129. Group by the code in field 9 of plate 1, visit 0 .....	149
2.130. Group by range of ages .....	150
2.131. Stats for all fields on plate 7 (for all visits) .....	151
2.132. Stats for all fields on plate 7 at visit 0 .....	151
2.133. Stats for fields 8 through 12 inclusive on plate 22 (for all visits) .....	151
2.134. Group plate 6 observations by sex (which is visit 0 plate 1 field 9) .....	151



# Preface

## 1. Getting Help

DF/Net Research, Inc.'s Technical Support Department can be contacted Monday to Friday between 9:00 am and 5:00 pm Eastern Time via any of the following methods:

55 Head Street, Suite 403  
Dundas, Ontario L9H 3H8  
Telephone: (905) 522-3282  
Email: <help@dfnetresearch.com>  
URL: [www.dfnetresearch.com](http://www.dfnetresearch.com)

## 2. 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;
```

If a line starts with # or %, this character denotes the system prompt and is not typed by the user.

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`.
- Menus and menu items are shown as: **File > Exit**.

# Chapter 1. Introduction

## 1.1. Restrictions

Permission to run reports from the UNIX command line is restricted to user `datafax` and to users in group `studies`. **DFExplore** users do not require a UNIX login account and are restricted only by the permissions specified in their roles.

Since database access restrictions may differ by study role and even for different users with the same role, execution of the same report by different users may lead to different results, and some users may be blocked from running a report because of inadequate permissions. Database access restrictions can be specified by site, subject, visit, plate and workflow level to limit the data records a user can read, and thus the output of each report. For example, clinical investigators allowed to see only the data for their own site will see results for their site alone when running the site and subject tracking reports. The table summarizes the affect of database access restrictions on each report.

**Table 1.1. Conformance with Access Restrictions for Standard Reports**

Conformance	Report Name(s)
(none): All database access restrictions are ignored. The output from this report is not affected by access restrictions and is the same for all users.	DF_ATcrfs, DF_ATfaxes, DF_ICcenters, DF_ICschema, DF_ICvisitdates, DF_ICvisitmap, DF_QCfaxlog, DF_QCstatus, DF_SScenters, DF_SSschema, DF_SSvars, DF_SSvisitmap, DF_WFcrfs, DF_WFcrfsperwk, DF_WFdiskusage, DF_WFfaxes, DF_WFqcs
(record): All database access restrictions are applied. The output is restricted to those records for which the user has access permissions. The output from this report will vary across users and will reflect their individual record access restrictions.	DF_ATmods, DF_ICkeys, DF_ICqcs, DF_ICrecords, DF_PTcrfs, DF_PTlist, DF_PTmissing, DF_PTqcs, DF_PTunexpected, DF_qcsbyfield, DF_stats
(subject): All database access restrictions are applied. The output is restricted to those subjects for whom the user has full visit and plate access permissions. The output from this report will vary across users and will reflect their individual subject access restrictions.	DF_PTvisits
(site): All database access restrictions are applied. The output is restricted to those sites for which the user has full subject, visit and plate access permissions. The output from this report will vary across users and will reflect their individual site access restrictions.	DF_CTcrfs, DF_CTqcs, DF_CTvisits, DF_PTschedule, DF_QCfax, DF_QCprint, DF_QCreports, DF_QCsent, DF_QCview
(full): All database access restrictions are applied. Only users with unrestricted access to all records in the study database can run this report. The output from this report will be the same for all such unrestricted users.	DF_ICimages, DF_QCupdate, DF_XXkeys, DF_XXtime

# Chapter 2. Reference Pages

This chapter is a reference for each DFdiscover and each legacy report. The reference information for each report is the same (except for formatting) as that available from **DFexplore's Reports** view. In that view, the information is available by selecting the report name and clicking **Explain**.

## 2.1. Organization of Reference Pages

The description of each report in this reference is divided into sections.

### 2.1.1. Name

The official name of the report.

The report name is case sensitive and must be typed exactly as specified. This section also provides the brief purpose of the report.

### 2.1.2. Synopsis

Lists all of the valid options for the report invocation.

When a report is invoked from the command-line, the study number is a required option and must appear as the first option. When the same report is invoked from Reports View in **DFexplore**, the study number is inserted automatically so it should not be manually inserted in the options list. The only situation in which the study number is not required, occurs when the user requests a usage message for the specified report. A usage message is produced when running a report from a shell or **DFexplore's Reports View** by specifying `-u` as the first option following the report name.

Each option is either optional or required. Most options are, not surprisingly, optional, a few are required, and yet others may require one option from a specific subset. Each option type is indicated with a particular notation. Except for some required options, a report option begins with `-` and is followed by an option letter. In some cases this may be enough to select the option; in other cases, the option letter will be further followed by an option string.

#### Option Letter

An option letter may be given different meanings in different reports. There is no requirement that the same option letter infers the same meaning when used across different reports.

**Table 2.1. Notation for report options**

Notation	Meaning
{study}	Required. The report will not function without the specification of this option. The option must be provided at the specified location in the list of options.
[-o]	Optional. The option may be omitted. The report will function, albeit differently, with or without this option.
[-o <i>string</i> ]	Optional, with an additional option string. The option may be omitted, however when it is used, the option letter must be immediately followed by a space and the option string. The option string generally ends at the next white-space character, however some option strings may contain spaces, and may or may not require that such strings be delimited by " .
{ [-s]   [-u] }	One of the options from the specified subset is required.
[ [-s #]   [-u #] ]	Several options are possible but only zero or one from the specified subset may be used.

### **2.1.3. Description**

Describes the report purpose in greater detail.

This section describes exactly how the report works in terms of environment, input requirements, output format, dependencies, etc. It may also include detailed information about the report behavior for commonly used combinations of options.

### **2.1.4. Options**

Detailed listing of available options, their use, and their meaning.

### **2.1.5. Examples**

Illustrates the report options and output by way of example(s).

### **2.1.6. See Also**

Lists other reference materials, typically other reports, that are relevant to the report. This is an optional section.

### **2.1.7. Limitations**

Describes any limitations in the use or capabilities of the report. This in an optional section.

## **2.2. Alphabetical Listing, DFdiscover Reports**

## Name

DF\_DatabaseDefinition — Database Definition

## Synopsis

DF\_DatabaseDefinition

## Description

Present detailed information of the plates and fields that define the study database. The presented information is a combination of visual plate layout, together with tabular information of the plate and field properties.

The visual presentation is linked in many places to facilitate easy navigation between plate, module and field level properties. information.

In the plate layout, each individual field is displayed in it's default on-screen position. Fields are grouped by layout to display module information. Fields belonging to the same module also share the same color.

Clicking in the background area of any plate updates the tabular information to display the values of each plate level property. Clicking any field updates the tabular listing to display the values of each field level property. Clicking in the background area bounded by a module layout area updates the display to a tabular listing of all instances of that specific module.

Navigation controls to navigate between plates are available via the [Previous](#) and [Next](#) controls or by selecting a plate from [Go To Table View](#).

## Options

None.

## Name


DF\_DataEntryTimeTrend — Data Entry Time Trend

## Synopsis

DF\_DataEntryTimeTrend [-site #]

## Description

Report the number of data records entered from paper and from EDC, and the number of elapsed days to complete data entry for each record. For time to complete data entry, the number of days between visit date and record creation date (delay) is calculated. If a record does not have a visit date, it is counted in the number of data records but it is not included in the number of days elapsed. The summary view presents the current, complete data for all permitted (or selected) sites on two axes, the Number of Records and Days Elapsed.

To view the change in data entry over time, check **Enable Timeline Animation** in the Options drawer. The report window updates to include animation controls, , and a timeline marked with the earliest date and the most recent date for which data is available. Use the controls to play, pause, and step forward/backward through the data.

Clicking on any bar switches to detail view for the selected site. In detail view, results are graphed for the site, month by month. For each month, four metrics are available: count of EDC records, count of paper records, average delay for data entry of EDC records, and average delay for paper records. The tooltip for each month shows both count and delay measures. Average is calculated, and presented, to one decimal place of precision.

## Options

**-site #** Report for the specified site only. By default, all permitted sites are reported.

## Name

DF\_Enrollment — Enrollment by Site

## Synopsis

DF\_Enrollment [-visit #] [-plate #]

## Description

Report enrollment numbers on a per site, or per country, basis. Enrollment numbers are aggregated on a monthly interval. Enrollment numbers are graphed showing the monthly enrollment and the aggregated, over time, enrollment.

By default, subjects are deemed to be enrolled if the plate containing the visit date for the baseline visit is present in the database. These visit and plate defaults can be changed via report options.

Output is presented in two views: summary and detail. In the summary view, current enrollment for all permitted sites is presented in a bar graph, one bar per site or country. The target enrollment for the site/country is optionally included. Clicking on any bar switches to detail view for the selected site/country. In detail view, monthly enrollment data is presented that includes the enrollment in that month (bar) and the cumulative enrollment up to and including that month (stepped line). Sloping rate lines are also included: constant rate (which assumes that total enrollment is achieved over the interval between site start and end dates) and trend line (linear interpolation of available monthly data). Vertical markers are included for Target Date (where the constant rate line meets total enrollment), Projected Date (where the trend line meets total enrollment), and optionally Actual Date (where total enrollment was achieved).

## Options

- visit #**      Override the visit number used in the determination of the enrollment visit. By default, the baseline visit is taken as the enrollment visit.
- plate #**      Override the plate number used in the determination of the enrollment date. By default, the baseline date is taken from the plate containing the field with the `VisitDate` style.

## Name

DF\_InboundTraffic — Inbound Traffic

## Synopsis

DF\_InboundTraffic

## Description

Report the volume of new data entering the database over time, categorized by data source, either CRF or EDC. The time interval can be one of week, quarter or year. Quarter is an approximation: weeks 1 through 13, inclusive, are quarter 1, weeks 14 through 26 are quarter 2, weeks 27 through 38 are quarter 3 and weeks 39 through 52 are quarter 4.

Volume is displayed as volume for the interval, as well as accumulated volume since start.

## Options

None.



## Name

DF\_PlateStatus — Plate Status

## Synopsis

```
DF_PlateStatus {-plate #}
```

## Description

Display simple counts and statistics for all fields on the requested plate. This report is similar to **DF\_stats** - the underlying data is the same but the presentation style is different.

The first part of the output is a summary table. Each data field is a row in the output table. The properties of each data field are presented in columns of the table. For each data field, the first 3 columns are properties defined by the database setup:

- Field Number
- Title
- Field Type

The remaining columns contain derived / calculated results summarizing the data values in the database. These results include:

- Totals: the number of records containing this field. This value should be the same for every field (row) in the table.
- Blank, Illegal, Invalid, Legal, Missing Value: counts of the number of records meeting each criteria. Each data value will meet only one criterion.
- Minimum, Maximum, Mean, Median, Mode Standard Deviation: statistical measures for the data field. These measures are reported for fields with a number field type. For fields with a field type of string, the length of the string is used in these measures.

The second part of the output is a simple chart view of the values meeting each of the data criteria. The length of each row in the output is the same as the Total value for the matching row in the table.

## Options

**-plate #**     Select the plate number (required)

## Name

DF\_QueryReport — Query Report

## Synopsis

```
DF_QueryReport [-site #] [-id #] [-internal]
```

## Description

This report provides a detailed listing of which required CRFs are missing, which scheduled visits are overdue, and which queries are unresolved and outstanding. It has the same purpose as Schedule View in **DFexplore** and the **DF\_QCreports** legacy report.

The output is split into 3 sections: the subject schedule, correction queries and clarification queries. The subject schedule is presented for each subject with at least one CRF in the database, unless one of the `-site #` or `-id #` options is also specified. If `-internal` is set, any queries identified as being for internal use are included in the correction and clarification queries lists.

Unlike **DF\_QCupdate**, this report does not insert missing page and overdue visit queries in the database. Instead it generates an up-to-date list each time that the report is run.

## Options

- site #** Report status and queries for the specified site
- id #** Report status and queries for the specified subject
- internal** Include internal use queries

## See Also

[DF\\_QCreports](#)  
[DF\\_QCupdate](#)

## Name


DF\_QueryStatusTrend — Query Status Trend

## Synopsis

DF\_QueryStatusTrend [-site #]

## Description

Report the number of resolved and outstanding queries on a per site basis. The summary view presents all permitted sites or a single selected site. For each included site, the current, total number of resolved and outstanding queries is presented.

To view the change in query status over time, check [Enable Timeline Animation](#) in the Options drawer. The report window updates to include animation controls, , and a timeline marked with the earliest date and the most recent date for which data is available. Use the controls to play, pause, and step forward/backward through the data.

Clicking on any bar switches to detail view for the selected site. In detail view, the number of resolved and outstanding queries is displayed over time, sampled on a monthly interval. Each query is attributed to the month in which it was created. Since the total number of queries continues to increase over time (resolved queries are typically not deleted), the detail view shows a comparison of resolved versus outstanding queries over time. Is the number (or percentage) of outstanding queries declining (as hoped) or increasing?

The output simplifies the presentation by including the status of a query in the sampled month - the output does not consider that a query could have had intermediate resolved status in an earlier month and may now be outstanding again.

## Options

**-site #** Report for the specified site only. By default, all permitted sites are reported.

## Name


DF\_QueryTimeTrend — Query Time Trend

## Synopsis

DF\_QueryTimeTrend [-site #]

## Description

Report the number of elapsed days to resolve queries. For time to resolve queries, the number of days between query creation and query resolution dates is calculated. If the query is currently outstanding, the report assumes that it will be resolved today and uses today as the resolution date. The summary view presents all permitted (or selected) sites on two axes, the Number of Queries and Days Elapsed.

To view the change in query resolution over time, check [Enable Timeline Animation](#) in the Options drawer. The report window updates to include animation controls, , and a timeline marked with the earliest date and the most recent date for which data is available. Use the controls to play, pause, and step forward/backward through the data.

Clicking on any bar switches to detail view for the selected site. In detail view, results are graphed for the site, month by month. Queries are attributed to the month in which they were created. For each month, three metrics are available: count of resolved queries, count of outstanding queries and average delay to resolve queries. The tooltip for each month shows both count and delay measures. Average is calculated, and presented, to one decimal place of precision.

## Options

**-site #** Report for the specified site only. By default, all permitted sites are reported.

## Name

DF\_QueryUpdate — Query Update

## Synopsis

DF\_QueryUpdate [-site #] [-id #]

## Description

This report applies the visit scheduling and CRF page requirement rules to determine which required CRFs are missing and which scheduled visits are overdue. It has the same purpose as Schedule View in **DFexplore** and the **DF\_QCupdate** legacy report.

The output lists each required CRF that is missing and each scheduled visit that is overdue. If the determination is influenced by a condition other than the study visitmap, the condition(s) are identified in the last 3 columns.

Unlike **DF\_QCupdate**, this report does not insert missing page and overdue visit queries in the database. Instead it generates an up-to-date list each time that the report is run.

## Options

- site #** Report missing pages and overdue visits for the specified site
- id #** Report missing pages and overdue visits for the specified subject

## See Also

[DF\\_QCreports](#)  
[DF\\_QCupdate](#)

## Name

DF\_SBhistory — History of Changes

## Synopsis

```
DF_SBhistory {-id #} [-visit #] [-plate #] [-field #]
```

## Description

DFdiscover maintains a change history of every data value or metadata value that is added/modified/deleted during the course of the study. This report outputs the change history in tabular format, for one specific subject ID. The output includes:

- identification of when the change was made (**Date and Time**), by whom (**User Name**) and the type of change (**Type**)
- identification of the value that was changed, including the data keys (**Subject, Visit, Plate**), the data field that was changed (**Data Field** and **Data Field Description**)
- the actual change (**What Changed, Old Value, New Value, Old Code Label, New Code Label**)
- and the related metadata (**Query Usage/Reason Text, Query Category/Reason Code, Status, Level**)

By default, all data and metadata changes are reported. To view only the changes to data, queries or reasons, select the desired view using **Change Records** in the Options drawer.

The report output can be quite lengthy if there are many data points collected per subject. To further filter the output, include only a specific visit (`-visit #`), plate (`-plate #`) or field (`-field #`). The filters are typically nested so that a specific visit can be included (`-visit #`), a specific plate within a specific visit (`-visit # -plate #`), or a specific field of a specific plate of a specific visit (`-visit # -plate # -field #`). However, it is also possible to view the change history for a specific field across the entire database, as well as other combinations of options.

It is possible to search the output by selecting the field of interest in the **Search by** dropdown list and then specifying the value to search for. Each column can be sorted in ascending or descending order by clicking the column heading. Each click reverses the previous sort order.

This report is similar in behavior to **Plate > List History of All Changes on this Page** (use `-plate #`) or **Field > List History of All Changes on this Field** (use `-field #`) available in other views.

## Options

- id #** Report change history for the selected subject ID (required)
- visit #** Filter to include only the selected visit number
- plate #** Filter to include only the selected plate number
- field #** Filter to include only the selected field number

## See Also

[DF\\_ATfaxes](#)  
[DF\\_ATmods](#)

## Name

DF\_SBSchedule — Subject Schedule of Visits

## Synopsis

DF\_SBSchedule [-site #] [-id #]

## Description

This report lists a summary of the subject schedule. The summary includes for each permitted subject ID:

- which visit was the entry, or start visit, including the date that the visit occurred
- which visit was the most recently completed visit, including the date that the visit occurred
- which visit, if any, is scheduled to occur next
- the status of the visit schedule for the subject, including outstanding queries if any or the number of days that the next visit is overdue

If the next visit is overdue, and there are outstanding queries, the overdue visit status is prioritized for display.

Any row in the tabular output can be clicked to present the detailed scheduling information for that particular subject ID. This detail includes the known visit schedule for the subject and includes the schedule date and status (incomplete, final, overdue and/or days off-schedule) for each visit (calculated from the available visit dates and the visitmap), and the actual dates of visits that have been received.

It is possible to search the output by selecting the field of interest in the [Search by](#) dropdown list and then specifying the value to search for. Each column can be sorted in ascending or descending order by clicking the column heading. Each click reverses the previous sort order.

## Options

**-site #** Report scheduling information for the specified site

**-id #** Report scheduling information for the specified subject

## See Also

[DF\\_QCreports](#)

[DF\\_QCupdate](#)

[DF\\_PTschedule](#)

[DF\\_SBunexpected](#)

## Name

**DF\_SBunexpected** — Subject Unexpected Visits

## Synopsis

DF\_SBSchedule [-site #] [-id #]

## Description

This report lists unexpected visits and plates found in the study data. It is possible to filter the listing to include only a specific site or specific subject ID; by default, all unexpected visits and plates for permitted data are reported.

Visits are unexpected when:

- they are excluded by a condition in the conditional visit map
- they are excluded by a condition in the conditional cycle map
- they have occurred after the date of the termination visit
- the visit schedule was aborted in a prior cycle

Plates are unexpected when:

- they are not defined in the visitmap as either required or optional
- they are excluded by one or more conditions specified in the conditional plate map
- a missed visit plate is present for the matching visit

The tabular output includes:

- the keys for the unexpected item (**Subject ID**, **Visit ID**, **Label**, **Plates**)
- indication of whether the visit or plate is unexpected (**Unexpected**)
- a text description of the problem, which may include references to defined conditions (**Problem**)

It is possible to search the output by selecting the field of interest in the **Search by** dropdown list and then specifying the value to search for. Each column can be sorted in ascending or descending order by clicking the column heading. Each click reverses the previous sort order.

## Options

- site #** Report unexpected visits and plates for the specified site
- id #** Report unexpected visits and plates for the specified subject

## See Also

[DF\\_QCupdate](#)  
[DF\\_PTschedule](#)  
[DF\\_SBSchedule](#)



## Name

DF\_SBAliasmap — Subject Alias Map

## Synopsis

DF\_SBAliasmap [-site #] [-id #]

## Description

This report displays in a simple tabular format the mapping of subject ID to subject alias. Options to filter by site or subject ID are available; with no options, the alias mapping for all subject IDs is listed.

## Options

- site #** Report aliases for all subjects at the specified site
- id #** Report the alias for the specified subject ID

## See Also

[DF\\_SBAliaslog](#)

## Name

DF\_SBAliaslog — History of Subject Alias

## Synopsis

```
DF_SBAliaslog ; [-site #] [-id #]
```

## Description

This report displays in a simple tabular format the history of updates in subject ID to subject alias mapping. For each update, the date and time are reported together with the name of the user making the update, the type of update (new, change, delete), the subject ID that is being updated, and the previous and new subject alias mapped by the subject ID.

## Options

**-site #** Report history for all subjects at the specified site

**-id #** Report history for the specified subject ID

## See Also

[DF\\_SBAliasmap](#)

## Name

DF\_SSsites — Site Definition

## Synopsis

DF\_SSsites

## Description

Present a tabular listing of detailed site information. The listing is a read-only presentation of site details defined in **DFsetup**.

The detailed site information is grouped for display purposes into:

- **Basic.** ID, Name, Subjects, Start Date, End Date, Enrollment Target, Test Only
- **Contact.** Contact, Address, Country Code, Fax, Telephone, Investigator, Reply To
- **Protocol.** Effective Date and Version, for each of up to 5 protocol pairs

If basic site information is included in the display, it is possible to search for specific site ID, Name, or Subject. When searching by subject, the search will search inside subject ID ranges.

## Options

None.

## Name

DF\_STcrfs — CRFs by Site

## Synopsis

DF\_STcrfs

## Description

Report and graph workflow statistics for time to receive data, enter data and save data as final. Statistics are grouped by site, or optionally by country.

An optional view, **Time to Final**, is available that presents the percentage of records that have final status on arrival (did not require further editing or clarification) and the percentage of records that have final status now.

Detailed numeric values for any site, or any country, can be obtained by clicking on the corresponding chart entry.

## Options

None.

## Name

DF\_STqueries — Queries by Site

## Synopsis

DF\_STqueries

## Description

Report summary information about queries metadata in the database. Summary information can be reported by site (default) or country. Summary information is presented in chart and table form. Information is similar to results produced by **DF\_CTqcs**.

Summary information is presented in 4 different views:

- **Queries by Status.** Query status is one of: Resolved Corrected Queries, Resolved Irrelevant Queries, Resolved Not Available Queries and Pending Queries
- **Categorical Age of New/Unsent Queries.** Age category of new queries is one of: <= 10 Days, 11-20 Days, 21-30 Days, 31-40 Days, 41-50 Days, 51-60 Days and > 60 Days
- **Categorical Age of Sent/Unresolved Queries.** Age category of sent but unresolved queries is one of: <= 10 Days, 11-20 Days, 21-30 Days, 31-40 Days, 41-50 Days, 51-60 Days and > 60 Days
- **Percent Resolved Queries.** As the study progresses, the percent of resolved queries approaches 100%.

## Options

None.

## Name

DF\_STvisits — Visits by Site

## Synopsis

DF\_STvisits

## Description

Report the number of visits, by visit number, that have occurred at each site.

A visit is deemed to have occurred if there is at least one plate in the database with that visit number. The results are similar to the results produced by **DF\_CTvisits**.

## Options

None.

## Name

DF\_Status — Status and Level Summary

## Synopsis

DF\_Status [-group]

## Description

Report summary information about the database status, summing over workflow level and grouping by the status of records, queries and reasons.

The summary information is similar to the information available in **DFexplore** via **View** > **Status**.

## Options

**-group** Group results by site, instead of the default, by workflow level

## Name

DF\_queriesbyfield — Queries by Field

## Synopsis

DF\_queriesbyfield

## Description

Tabulate and graph the number of queries, of each category, for each field in the database. The report purpose and numeric values are the same as the legacy report **DF\_qcsbyfield** - the presentation however is different.

The output includes each data field that has at least one query. By default, output is sorted by increasing plate number, and field number within plate. Fields which have no queries are not presented.

Choosing **Sort by Total** produces output ordered by the most frequently queried data fields.

Note that missing plate and overdue visit queries are always attributed to the subject ID field on a plate.

## Options

None.

## 2.3. Alphabetical Listing, Legacy Reports



## Name

**DF\_ATcrfs** — Track keys associated with a CRF page over time

## Synopsis

```
DF_ATcrfs {study} [-all] {-f CRFpageID}
```

## Description

The study number and key fields (ID, visit, plate) associated with a CRF page may change over time, because the key fields were entered incorrectly and later fixed, or because the CRF page was moved to a different study. **DF\_ATcrfs** tracks these changes by searching the audit trail of the current study, or optionally all studies registered in DFdiscover, for all journal records related to a specified CRF page. If the option to search all studies is enabled, all known studies are searched, whether the study is currently active or not.

For each journal record identified, **DF\_ATcrfs** displays:

- The database transaction date in yyyy/mm/dd format.
- The database transaction time in hh:mm:ss format.
- The study number of the database in which the journal record was found.
- The study number, from field 4 of the journal record. <sup>1</sup>
- The subject ID from field 7 of the journal record.
- The visit number from field 6 of the journal record.
- The plate number from field 5 of the journal record.
- The validation level (0-7) from field 2 of the journal record.
- The record status (new,final,incomplete,pending,FINAL,INCOMPLETE,PENDING or delete) from field 1 of the journal record.
- The login name of the user who performed the database transaction.

In addition to showing the above information from the journal files, **DF\_ATcrfs** also checks the router and prints a message at the end of the report indicating that the CRF page is currently located in the router, if that is the case. A CRF page will be placed in the router on arrival if the barcode could not be read. It will also be moved to the router from a study database if **Change Keys** (in **DFexplore**) is used to change the study number. When this happens the data record and any metadata records (queries and/or reasons) are deleted from the original study database, and the CRF image starts over when it is routed to the new study database.

## Options

**-all** Search all DFdiscover study databases, not just the current one. All known studies are searched, whether the study is currently active or not. This generates exhaustive output but at the expense of report speed.

---

<sup>1</sup> In almost all cases, this study number value will be identical to the study number value in the previous column. The two values will differ only when a database is copied, outside of DFdiscover, from one study number to another. In this case, the original study will report identical study number values in the two columns, while the copied study will have a database study number (the new study number) which differs from the study number found in the journal record (the old study number).

**-f CRFpageID** The identifier for the CRF page to be tracked (required). The *CRFpageID* must be specified in the standard *YYWW/SSSSPPP* format.

## Examples

### Example 2.1. Report the audit trail for CRF page 1525/0016001

```
DF_ATcrfs: Track keys associated with a CRF page over time. Jul 28,2015 13:04
```

```
Keys history for CRF page 1525/0016001
```

yyyy/mm/dd	hh:mm:ss	DBN	study	id	visit	plate	level	status	user
2015/06/21	14:25:59	253	253	1019	0	1	0	new	datafax
2015/06/21	14:49:51	253	253	1019	0	1	1	final	user1
2015/07/28	13:02:17	253	253	1019	0	1	2	delete	user2
2015/07/28	13:03:28	252	252	1019	0	1	0	new	datafax
2015/07/28	13:04:16	252	252	1019	0	1	1	final	user2

The output shows that CRF page 1525/0016001 was originally entered into study 253 on June 21, 2015. Then on July 28, 2015 it was moved from study 253 to study 252. The record with status delete was created when **Change Keys** was used in **DFexplore** to move the CRF page to the router. The subsequent record with status new was created when the CRF page was identified in **DFexplore** as belonging to study 252. Note that new records, arriving at level 0 and attributed to user `datafax`, indicate that the CRF page has been placed in the new record queue to await validation by data management staff.

## Limitations

**DF\_ATcrfs** is really an administrative report and for maximum benefit should be restricted to user `datafax`.

- **DF\_ATcrfs** determines the history of the CRF by reading the journal records from the requested studies. If any journal record cannot be read due to permission problems, it will be quietly skipped. This may result in incomplete information concerning the history of the CRF.
- **DF\_ATcrfs** will report all known keys history information, independent of any access permission restrictions placed on the current user.

## See Also

[DF\\_ATmods](#)  
[DF\\_ATfaxes](#)

## Name

**DF\_ATfaxes** — Trace validation history of each page in selected documents

## Synopsis

```
DF_ATfaxes {study} { [-t yy/mm/dd-yy/mm/dd] | [-t ] | [-f documentname] }
[-s sender] [-j 0123] [-pg rows columns]
```

## Description

**DF\_ATfaxes** displays the validation history of each page in selected documents. The output includes:

- Document arrival: date, time, sender identification, page number
- Data Record Keys: id, visit, plate
- Journal: status, validation level, date, time, user

**DF\_ATfaxes** proceeds in 2 steps.

### 1. Read the DFdiscover fax\_log.

This step selects documents by:

- arrival date or date range, `-t`, and/or,
- document name, `-f`, and optionally also by
- document sender ID, `-s`

and shows all documents that meet the selection criteria, from all DFdiscover studies.

### 2. Read the study journal files.

This step is performed only if `-j` is selected. The journal files for the current study are read to locate all journal entries for all pages in the documents selected by the first step. Only those documents that are journaled in the current study are displayed.

Document arrival dates can be specified in either `yy/mm/dd` or `format`. If the former is used, the study start date (recorded in `DFschema`) is used as the pivot year for date imputation. A single date may be specified to select documents received on that date, or a date range may be specified to select documents received between the two dates, inclusive.

Documents may also be selected by document name using `-f`, such as `-f 1805/0012` which selects the 12<sup>th</sup> document received in the 5<sup>th</sup> week of 2018, and `-f 1752/0001` selects the 1<sup>st</sup> document received in the 52<sup>nd</sup> week of the year 2017. Every document received by DFdiscover is given a unique name, composed from the year, week, and document number. Pages entered by EDC data entry also use this naming scheme, thus gaps may appear in the document names retrieved from `fax_log`.

When `-f` is used only one document name may be specified, but the match is performed anywhere in the document name, thus `-f 1805` will match all documents received in the 5<sup>th</sup> week of 2018 (and the 1805<sup>th</sup> document received each week, if any), and `-f 1805/004` will match image numbers 0040 through 0049 of the 5<sup>th</sup> week of 2018. Be aware that a pattern like `-f 0001` will match all documents received in the first week of 2000 and all first documents received in any week.

Fax machines include a sender ID, if programmed by the sending site, which identifies the fax machine that sent the fax. For emailed documents, the sender ID is the sender's email address. For documents sent with **DFsend**, the username of the sender is the sender ID. This is stored in the `fax_log` and can be used to select documents. If the sender ID includes spaces

it must be enclosed within double quotes, e.g. `-s "555 111 1234"`. The match on sender ID is performed anywhere in the sender ID string. Thus `-s 555` will also match this sender, as well as any other sender with 555 anywhere in the sender ID.

The `-j` option is used to display journal entries for all CRF pages in the selected documents, but only those documents that contained CRFs for the current study will be in the study journals, thus only these documents will be displayed when `-j` is used. Output can include all or selected journal entries. Any combination of the selectors 0, 1, 2, and 3 can be specified with `-j`. Use:

- 0 to select the ICR step,
- 1 to select the first validation of the page,
- 3 to select the last validation of the page, and
- 2 to select all validations between the first and last.

For example:

- `-j 1` shows only the first record validation for each page, which is the entry from the new image queue,
- `-j 3` shows only the last validation for each page, which corresponds to the current status of the data record in the database,
- `-j 13` selects both first and last journal entries,
- `-j 123` selects all validation journal entries, and
- `-j 0123` selects all journal entries including the ICR step.

When `-j` is not used, the document name, number of pages in the document, arrival date, time and sender ID are displayed for the selected documents from the `fax_log`, and will include documents from all DFdiscover studies being conducted.

When `-j` is used, the status of each journal entry is coded as a single letter key, from the following legend:

```
Primary Records:  f=final, i=incomplete, p=pending
Deleted Records:  -=deleted
```

The status is followed by the validation level, where 0 represents the ICR step.

## Options

- `-t yy/mm/dd-yy/mm/dd-t` Select documents by the date period during which documents arrived (`-t`) and/or by their name (`-f`). At least one of these options is required.
- `-f documentname`
- `-s sender` Select documents from specified sender
- `-j 0123` Include information from journal records. Valid selectors include one or more digits from: 0=ICR, 1=1st entry, 2=middle entries, and 3=last entry
- `-pg rows columns` Set output page size in rows and columns (default is 56 rows by 85 columns)

## Examples

### Example 2.2. Show all documents received on Jan 16, 2018

```
-t 980116
```

```

Reading fax log
DF_ATfaxes: Audit trail for documents.  DFstudy 7.  Feb 10,2018 23:03      PAGE 1

document 1803/0024  pages 1      2018/01/16 00:01:34  sender: 61 3 93422947
document 1803/0025  pages 7      2018/01/16 09:35:45  sender: 215 402 9234
document 1803/0026  pages 1      2018/01/16 10:18:16  sender: 905 546 1231
document 1803/0027  pages 3      2018/01/16 10:46:59  sender: 715 387 7317
    
```

The output shows 4 documents, with ids 1803/0024 to 1803/0027, received between 1:34 AM and 10:46 AM on Jan 16, 2018 from 4 different senders. Note that we cannot determine which study each document belongs to.

### Example 2.3. Show all journal entries for the pages in document 9803/0027

```

-f 9803/0027 -j 0123

Reading fax log
Reading journals
DF_ATfaxes: Audit trail for documents.  DFstudy 7.  Feb 10,2018 23:04      PAGE 1

document 1803/0027  page 1/3      2018/01/16 10:46:59  sender: 715 387 7317
 65565      4    7    0  2018/01/16 10:49:40
 65565      4    7    i1 2018/01/16 12:41:55  jack          ❶
 65565      4    7    f2 2018/01/16 15:43:37  jane          ❷
 65565      4    7    f3 2018/01/18 11:52:49  mary          ❸

document 1803/0027  page 2/3      2018/01/16 10:46:59  sender: 715 387 7317
 65565      4    8    0  2018/01/16 10:49:40
 65565      4    8    i1 2018/01/16 12:42:02  jack          ❹
 65565      4    8    f2 2018/01/16 15:43:06  jane          ❺
 65565      4    8    f3 2018/01/18 11:52:43  mary          ❻

document 1803/0027  page 3/3      2018/01/16 10:46:59  sender: 715 387 7317
 65565      4   14    0  2018/01/16 10:49:40
 65565      4   14    i1 2018/01/16 12:42:07  jack          ❼
 65565      4   14    i2 2018/01/16 15:45:41  jane          ❽
 65565      4   14   -3 2018/01/18 11:53:24  mary          ❾
    
```

All 3 pages in the document are for subject 65565, visit 4, plates 7,8 and 14.

- ❶❷❸ All 3 pages were validated to level 1, incomplete by jack on Jan 16,2018.
- ❹❺❻ Later that day Jane raised all 3 records to level 2. Plates 7 and 8 were signed off final and plate 14 was signed off incomplete.
- ❼❽❾ On the 18<sup>th</sup> of January, Mary raised plates 7 and 8 to final at level 3, and deleted plate 14. It is possible that Mary selected another document to become the primary record for plate 14. To examine this we could run **DF\_ATmods** on plate 14, visit 4 for subject 65565.

The title identifies DFdiscover study number 7 as the current study. No output would have been produced if document 9803/0027 did not belong to study 7.

### See Also

[DF\\_ATmods](#)

## Name

**DF\_ATmods** — Trace database modifications over a time period

## Synopsis

```
DF_ATmods {study} {-t } [-w time1-time2] [-u user1,user2] [-f #] [-v #] [-l #, #-#] [-S #, #-#] [-P #, #-#]
[-din #, #-#] [-dex #, #-#] [-qin #, #-#] [-qex #, #-#] [-qprob #, #-#] [-d new,mod,del,none,all ] [-q
new,mod,del,in,ex,none,all ] [-N data,qcs ] [ [-by case] | [-by time] ] [-test]
```

## Description

**DF\_ATmods** creates an audit trail report showing modifications (additions, changes, and deletions) made to data fields, queries and reasons over time.

A full trace of all changes made to a study database can generate thousands of pages of output. Using the available options to limit the report to show only those changes made during a short time period, or by specified users, or to data for specified subjects, visits and plates, is highly recommended.

The report's format is illustrated in the following example output:

```
DF_ATmods: Database Changes For Study 253 Jul 25,2016 16:56 PAGE 1 ❶
      -din 8-22 -N data -t 20160725 -I 1003 -P 2 ❷
-----
ID=1003 SEQ=1 PLT=2 ❸
2016/07/25 13:52:55 jack DATA: NEW RECORD at level 1, incomplete ❹
  8. Patient Initials = CCC
     REASON new at level 1: Set by edit check SetInitials
  9. Entry Date = 03/07/06
 10. Does pt. meet eligibility = 2 (Yes)
 11. Eligibility: Explain = blank
 12. Medication Code # = 2543
 13. Date of Birth = 15/06/44
 14. Weight in kg = 075.3
     REASON new at level 1: Set by edit check CalcWeight
 15. Weight in lbs = 166.0
 16. Height in cm = blank
 17. Height in in = blank
     Query new at level 1: external, missing, status = new
 18. Pulse beats/minute = 062
 19. Smoking Status = 1 (Never smoked)
 20. Exercise = 2 (1-2 times/week)
 21. Date of First Follow-up = blank
     REASON new at level 1: a date was not set, patient will call back to make
     an appointment
 22. DFdiscover Screen Status = 2 (incomplete)

2016/07/25 14:53:21 susan DATA: MODIFIED at level 1, final ❺
 16. Height in cm: blank -> 180
     REASON new at level 1: Set by edit check CalcHeight
 17. Height in in: blank -> 071.0
     Query modified at level 1: external, missing, status = resolved corrected
```

```

Query Status: 1 (new) -> 5 (resolved corrected)
21. Date of First Follow-up: blank -> 03/08/06
REASON modified at level 1: a date was not set, patient will call back to
make an appointment -> appointment date has been set.

2016/07/25 16:43:51 bill          DATA: NO CHANGE      6
21. Date of First Follow-up:
REASON approved at level 1: a date was not set, patient will call back to
make an appointment

```

- ❶ The report title, repeated at the top of each page, identifies the study number, and the date and time the report was executed.
- ❷ The options used are identified below the report title.
- ❸ The database record to which modifications were made is identified by its 3 primary keys: the subject number (ID), the visit or sequence number (SEQ), and the plate number (PLT).
- ❹ Each transaction for this case is identified by the date, time and user who made the modifications, followed by the transaction type, which indicates whether the data record was new, modified, deleted or unchanged.

This is followed by a listing of the data fields, identified by number and descriptive label, for which modifications were made to the data value, queries and/or reasons.

For the transaction shown in this example all new data values entered for fields 8 to 22 are shown, due to options: -din 8-22 -N data. Without these options, only changes made after the initial data entry would have been shown.

New and/or modified metadata (queries and reasons) are shown directly below the data field with which they are associated. The output for each transaction does not show metadata that existed but did not change during the transaction.

- ❺ This transaction shows modifications made by susan about an hour after the new data record was created by jack. Susan entered the missing value for height that resulted in the resolution of the missing value query on that field. She also entered the date of the first follow-up appointment along with a reason for the change to that field.
- ❻ The last transaction for this case shows bill's approval of the reason entered by susan for the change to the date of first follow-up. For each transaction, the workflow (or validation) level at which modifications are saved is shown separately for the DATA, QUERY and REASON records.

## Options

- t Include only changes made between the specified dates (required). Only one date or date range may be specified. The keyword `today` can be used to identify the date on which the report is run. To include only changes made on a single date, specify only that date.
- w time1-time2 Include only changes made between the specified times of the day. Only one time or time range may be specified. Times must be specified in military (24 hour) notation (`hhmmss`), and may optionally include `:` as a delimiter.
- u user1,user2 Include only changes made by the specified user(s)
- f # Include only changes that occurred after record reached the specified validation level. Only one validation level may be specified. This option restricts the output to modifications made after a data record or query has been committed at or beyond the specified validation level. For example, `-f 6` would include modifications to data records after the first sign off at validation levels 6 or 7, and would show any modifications to queries and reasons that occurred after they were first committed at levels 6 or 7.

The first commit is not reported but all subsequent modifications including those made at the specified validation level will be reported.

- This option will show records and queries which were deleted after reaching the specified validation level, but the delete action ends the trace and the report will not show any subsequent changes for that case; i.e. if another case is created with the same keys it too would have to reach the fence before any subsequent changes to it would be reported.
- v #** Include only changes that occurred at the specified validation level. This option restricts the output to include just those data records, queries and reasons that were created, modified or deleted at the specified validation level(s). Remember that the validation level of data records, queries and reasons may be different. Thus restricting the output to changes made at a specified validation level will show changes to data fields without showing changes that were made to queries or reasons in the same transaction if the changes to queries and reasons occurred at different validation levels.
- l #, #-#** Include only the specified subject IDs
- S #, #-#** Include only the specified sequence or visit numbers
- P #, #-#** Include only the specified plates
- din #, #-#** Show modifications made only to the specified data field numbers. It is not required that the listed fields exist on all plates. A description of data field numbers can be obtained by running **DF\_SSchema** or **DF\_SSvars**.
- dex #, #-#** Exclude modifications made to the specified data field numbers. Fields listed are not included in the report. Exclusion has priority over inclusion; thus if a field is specified in both the `-din` and `-dex` options, the field is excluded.
- qin #, #-#** Show modifications made only to the specified query field numbers. A description of query fields can be obtained by running **DF\_SSchema** with the `-P 511` option.
- qex #, #-#** Exclude modifications made to the specified query field numbers. Fields listed are not included in the report. Exclusion has priority over inclusion so that if a field is specified in both the `-qin` and `-qex` options, the field is excluded.
- qprob #, #-#** Show modifications made only to the specified query categories from the list: 1=missing, 2=illegal, 3=inconsistent, 4=illegible, 5=fax noise, 6=other, 23=edit check missing page, 30-99=user-defined problem type. The creation and deletion of missing plate (code 21) and overdue visit (code 22) queries by **DF\_QCupdate** is not displayed by **DF\_ATmods**. Note that in case of changes to the query category labels over the course of the study, the current label as defined in **DFsetup** will be displayed, not the label defined at the time the query was added.
- d new,mod,del,none,all** Include only specified types of data changes: `new`=new data records, `mod`=changes to existing data records, `del`=deleted data records, `none`=do not show any data changes, `all`=all changes (default) which is equivalent to `new,mod,del`
- q new,mod,del,in,ex,none,all** Include only specified types of query changes: `new`=new queries, `mod`=changes to existing queries, `del`=deleted queries, `in`=internal queries, `ex`=external queries, `none`=do not show any query changes, `all`=all changes to queries (default) which is equivalent to `new,mod,del`
- N data,qcs** When new data records and/or queries are created, display the fields specified with the `-din`, `-qin`, `-dex`, and `-qex` options. By default the report only shows changes to existing data and metadata. Thus when new data records and queries appear in a transaction, the values of the data fields and query fields are not shown unless this option is used.



- by case-by time**                      Sort the output by time (default) or, first group by case (i.e. by each unique combination of the key fields id, visit, and plate) and then sort by time within case.
- test**                                      Test mode. Display the chosen options and exit.

## Examples

The examples demonstrate how different combinations of options can be used to filter the audit trail information for particular purposes.

**Example 2.4. Show new data records, modifications to existing records, and all internal and external queries added or modified, between Dec 1,2017 and today's date**

```
-d new,mod -q new,mod -t 20171201~today
```

**Example 2.5. Show modifications made to existing external queries by jane between Dec 1 and Dec 24, 2015**

```
-u jane -t 20151201~20151224 -d none -q mod,ex
```

To ensure that the output includes only query modifications, and does not also include data modifications, `-d none` is required.

**Example 2.6. Show all modifications, excluding queries, made after a data record reached validation level 6**

```
-v 6 -d mod -q none
```

Note that changes would include those that resulted from a re-submit that was processed at lower levels after the data record reached validation level 6.

**Example 2.7. Show all modifications made to the database between 8am and 9am this morning**

```
-t today -w 080000~090000
```

**Example 2.8. Show modifications made during the year 2015 to fields 15 through 18 and 22 of external queries**

```
-t 20150101~20151231 -d none -q mod,ex -qin 15~18,22
```

To also show these fields in new external queries created in the year 2015, add `-q new` and `-N qcs` giving:

```
-t 20150101~20151231 -d none -q new,mod,ex -qin 15~18,22 -N qcs
```

**Example 2.9. Show modifications made at validation levels 4 through 7, between July 1, 2015 and today's date**

```
-v 4~7 -t 20150701~today
```

**Example 2.10. Show modifications made after validation level 6, by jane, between Jan 1, 2015 and today's date**

```
-f 6 -u jane -t 20150101~today
```

This includes changes to data records and metadata records after they were committed at levels 6 or 7. Remember that data, queries and reasons all have separate validation levels. Thus the changes displayed for a given data field may show changes to the data value but not changes that occurred during that transaction to queries and/or reasons, and vice versa, if the data and metadata records have reached different validation levels.

## See Also

[DF\\_ATfaxes](#)

## Name

**DF\_CTcrfs** — Summary report of records received from each site

## Synopsis

```
DF_CTcrfs {study} [-x] [-c #, #-#] [-S #, #-#] [-P #, #-#] [-i rom] [-R]
[[-t yy/mm/dd] | [-t ] | [-t yy/mm/dd-yy/mm/dd] | [-t ]]
```

## Description

**DF\_CTcrfs** reports a summary of the time required for database processing of records received, during a specified time period, from each study site including the error monitor site. Time periods can be selected:

- from the beginning of the study
- from a specified date
- between two specified dates

Only visits and plates defined in the study visit map are included in this report.

The tabular report displays for each site (row) columns that contain:

- **Activity (since the beginning of the study or since the specified date).** The activity is reported by:
  - Number of subjects: number of subjects seen for any visit
  - Number of visits: number of visits for all subjects
  - Number of records: CRF pages received (only primary copy is counted)
- **Time (mean days) from visit date to first document transmission.** If re-submits exist for a plate only the first arrival (earliest document date) is used<sup>1</sup>.
- **Time (mean days) from visit date to data entry (initial record creation).** The number of days between the subject visit and the date that the initial data record was created for that CRF, regardless of how many re-submits may have occurred<sup>1</sup>.
- **Time (mean days) from visit date to a final database record.** If re-submits exist only the primary "final" record is used. If the primary record is not yet a "final" record, the time from the visit date to today's date is used, and a + is appended to the displayed average to indicate that it is based on incomplete data, and would be at least the value shown<sup>1</sup>.
- **How clean is the database?** The percentage of records that were final on first arrival and the percentage of records that are final now form the last 2 columns of the report. Clean on first arrival is determined by comparison of the record creation and last modification dates. If a record reaches final status on the same day it was first created, and the data on the record is not subsequently modified, it is considered to be "final on arrival" (Note: changing the validation level on a data record does not result in a change in the modification date stamp unless data fields are changed as well).

## Options

- c #, #-#                      Include only the specified site IDs.
- S #, #-#                      Include only the specified visit or sequence numbers.

<sup>1</sup>The number of records used in this calculation may be less than the total number of primary records received if some of the records do not have a visit date.

<b>-P #, #-#</b>	Include only the specified plates.
<b>-i rom</b>	Include only the specified types of plates; <i>r</i> =required, <i>o</i> =optional, <i>m</i> =missed visit
<b>-R</b>	Exclude EDC data entry and imported data. This option results in exclusion of records which were not submitted externally, that is, any EDC data entry records and any imported records in which a null image name (i.e. 0000/0000000) has been used.
<b>-t yy/mm/dd-t</b>	Include records in visits occurring since specified date, inclusive of the specified date
<b>-t yy/mm/dd-yy/mm/dd-t</b>	Include records in visits occurring between specified dates, inclusively

## Limitations

**Only visits & plates defined in the visit map are included.** This report focuses on time since each subject visit and thus only includes visits and plates that are defined in the study visit map. Both required and optional plates defined for each visit (or sequence) number are included.

**Only records with valid dates are used in the time calculations.** The denominator for the mean time between visit and image arrival requires both a visit date and a image arrival date. If either is unavailable that record will not be included in the calculation of the mean. If the site has no subjects with usable visit dates, N/A is output for the `DB Entry` and `DB Clean` columns. Similarly, if none of the CRFs for the current site can be found in `/opt/dfdiscover/work/fax_log`, N/A is output for the `Fax In` column.

## See Also

[DF\\_XXtime](#)

## Name

**DF\_CTpages** — Summary report of CRF pages received during a specified time period.

## Synopsis

```
DF_CTpages {study} {-t yy/mm/dd-yy/mm/dd} [-i #, #-#] [-c #, #-#] [-v #, #-#] [-p #, #-#] [-R] [-x]
```

## Description

**DF\_CTpages** reports the number of data records in the database that arrived within a specified time window (i.e., between 2 dates). The document arrival date is represented by the arrival date stored in the DFdiscover faxlog file. If an arrival date is not available, the record creation date is used. This should only apply to EDC data entered records and records that have been imported (using **DFimport.rpc**) without a matching image in the faxlog, but could also be applied if the faxlog is incomplete.

**DF\_CTpages** gets its information from the DFdiscover timing summary files and `DFX_dfin`. If necessary, **DF\_CTpages** first executes **DF\_XXtime** to update these 2 files.

The output from **DF\_CTpages** includes 6 columns of data: 3 for primary records (`final`, `incomplete`, `pending`), 1 for all secondary records, 1 for returned Query Reports (plate 501), and 1 for the total number of records present in the previous 5 columns. Missed records are not counted, because they simply identify pages that did not arrive and are no longer expected to arrive.

New records, which have not yet been entered into the database, and records sitting in the router are not included in the report, because the record keys (ID, visit and plate numbers) and site ID have not yet been verified and are thus not known with certainty. However, **DF\_CTpages** does print warning messages at the bottom of the report if any records are found in the new record queue and/or in the router that arrived during the specified time interval. Since the record keys and site ID are not known with certainty, any related filter options are not applied when counting these records.

## Options

<b>-t yy/mm/dd-yy/mm/dd</b>	Include pages received in the specified date range (required)
<b>-i #, #-#</b>	Include only the specified subjects
<b>-c #, #-#</b>	Include only the specified sites
<b>-v #, #-#</b>	Include only the specified visit/sequence numbers
<b>-p #, #-#</b>	Include only the specified plate numbers
<b>-R</b>	Exclude pages that are not registered in the faxlog. Typically this will only exclude EDC data entry records and records imported using , but it could also exclude submitted pages if the faxlog has been truncated, or is incomplete for any other reason.

## Limitations

- Only pages that have been processed to validation level 1 or above can be counted, thus to obtain accurate results from **DF\_CTpages**, the new record queue and the pages in the **DFexplore** router need to be emptied.
- Any pages that are no longer in the database, either because they were deleted or routed to a different study database, will not be included in the **DF\_CTpages** output.

## See Also

[DF\\_XXtime](#)

## Name

**DF\_CTqcs** — Summary report of external queries for each site

## Synopsis

```
DF_CTqcs {study} [-x] [-a] [-v] [-c #, #-#] [-P #, #-#] [-S #, #-#]
[[-t yy/mm/dd] | [-t] | [-t yy/mm/dd-yy/mm/dd] | [-t]]
```

## Description

**DF\_CTqcs** reports a summary of the user-created, external queries added during data validation. queries are selected by their creation date from a specified time period that may include one of:

- from the beginning of the study
- from a specified date
- between two specified dates

The following queries are not included in the report:

- Internal queries
- Missing page queries (added by **DF\_QCupdate** or edit checks),
- Overdue visit queries (added by **DF\_QCupdate**)

The summary is in the form of two tabular reports. The first report displays for each site (row), columns that contain:

SITE	TOTAL RECORDS	TOTAL QRYS	RATE/100 REC	RESOLVED #	QRYS %	STATUS & TIME TO RESOLUTION	Irrel. DAYS	Pending
	①	②	④	⑤		Correct N.A.	⑥	⑦

- ① Number of primary records created
- ② Number of external queries created during validation
- ③ Query rate = number of queries created per 100 primary records created
- ④ Number and percentage of these queries which have been resolved
- ⑤ Number of each resolution type from the list:
  - corrected
  - not available
  - irrelevant
- ⑥ Mean time (in days) from the creation of the queries to their resolution
- ⑦ Number of queries that have replies that are pending review

The second table displays for each site (row), columns that contain:

SITE	AGE (DAYS) NEW/UNSENT QUERIES①					AGE (DAYS) SENT/UNRESOLVED QUERIES②				
	00-10	11-20	21-30	31-60	> 60	00-10	11-20	21-30	31-60	> 60

- ❶ Categorical time (in days) from query creation to today's date, for unresolved queries that have not been sent to the sites in a Query Report.
- ❷ Categorical time (in days) from query creation to today's date, for unresolved queries that have been sent to the clinical sites. Queries with status pending are included in this category.

An alternative report format can be requested with `-a`. This report format is a single table that is a hybrid of the two, already described, tables. Rather than displaying the status of resolved queries from the first table, the hybrid report displays the age of unresolved queries from the second table, so that the column headings read:

SITE	TOTAL	TOTAL	RATE/	RESOLVED	QRY	AGE (DAYS) OF UNRESOLVED QRY				
	RECORDS	QRY	100 REC	#	%	<=10	11-20	21-30	31-60	>60

**DF\_CTqcs** uses summary file which is created by **DF\_XXtime**. **DF\_CTqcs** checks the summary file `DFX_time1` against the study database and sites database to see if either database has changed since the file was last created. If either or both databases have changed, **DF\_XXtime** is executed to rebuild `DFX_time1`, otherwise this step is skipped. If **DF\_CTqcs** is run with the option `-x`, the results will reflect the state of the database on the last time that **DF\_XXtime** was run.

If subject IDs exist in the database which do not correspond to any of the sites in the sites database, they are assigned to the ERROR MONITOR. By default, **DF\_CTqcs** includes such subjects when making this report. If `-v` (for verbose mode) is used, such cases will be listed at the end of the report.

## Note

The count of primary records and queries created during the specified time period are not necessarily related, as some or all of the selected queries may have been added to previously created records (which would not be selected). Thus this report may reflect both coordinating site activity as well as site performance. An increase in querying by data staff could make it appear that CRF problems are rising when it is really just an increase in the search for problems. Also an increased focus on some sites could make the quality of CRFs arriving from those sites appear worse than that from sites under less scrutiny. However under most circumstances the relative number of queries and primary records created during the specified time period should generally reflect the quality of the original CRFs being received from each site.

## Options

- `-a` Show age of unresolved queries (instead of query status).
- `-v` Verbose. Subject IDs not belonging to any site are reported at the end of the report.
- `-c #, #-#` Include only the specified site IDs.
- `-P #, #-#` Include only the specified plates.
- `-S #, #-#` Include only the specified sequence or visit numbers.
- `-t yy/mm/dd-t` Include records and queries created since specified date, inclusive of the specified date
- `-t yy/mm/dd-yy/mm/dd-t` Include records and queries created between specified dates, inclusively

## Limitations

1. This report cannot be used to obtain the status of internal queries or external queries created for missing pages and overdue visits.
2. Mean days from creation to resolution is based on those queries that have been resolved so far and thus may not reflect the true average time if there are many unresolved queries of long duration.

## See Also

[DF\\_XXtime](#)

[DF\\_PTqcs](#)



## Name

**DF\_CTvisits** — Number of subjects who have reached each visit at each site

## Synopsis

```
DF_CTvisits {study} {-S #, #-#} [-x] [-c #, #-#] [[-w all] | [-w VisitDates]] [[-l #]] [-n #]
```

## Description

**DF\_CTvisits** tabulates, by site, the number of subjects who have reached each specified visit. The list of visits to tabulate is given by the required `-S #, #-#` option. From that list, all visits that have been registered in the database with status final, incomplete, missed or pending are included if `-w all` is specified; or only visits with a legal `Visit Date` are included if `-w VisitDates` is specified (default).

## Options

- S #, #-#**            Include only the specified visits (required)
- x**                    Do not execute **DF\_XXkeys**. Reported information will reflect what was in the database the last time that **DF\_XXkeys** was executed
- c #, #-#**            Include only the specified sites. By default, all sites including the error monitor are included.
- w all**  
**VisitDates**            Include all visits (`-w all`) or only visits with legal `VisitDates` (`-w VisitDates`, default)
- l #**                    Select the field from the site record to use for display, following the site ID, of a site label. The field numbers used in site records are described in [Programmer Guide, DFcenters - sites database](#)
- n #**                    Format the output to include at most the first # characters (default is 0) of the site label selected by `-l #`

## Examples

**Example 2.11. report on sites 1-20 for visits 0-2,4 labeling each site with the first 4 characters of the site name (which is in field 3 of the site record)**

```
-c 1~20 -S 0~2,4 -l 3 -n 4
DF_CTvisits: Study Visits Received by Site.            DFstudy 7. Apr 22,2017 23:21

DF_CTvisits  Number of Visits Received for Study Example
              only visits with legal VisitDates are included

Site            0        1        2        4
1 Vanc            21      21      21      21
2 Calg            2        2        2        2
3 St.            16      15      15      15
4 Univ            19      16        9      10
5 Roya            16      16      16      16
6 Sout            16      16      16      16
9 Sieb            28      27      27      26
```

10 Univ	58	57	57	53
11 Sunn	18	18	18	16
13 The	43	42	42	39
14 McMa	32	32	31	31
16 Otta	29	29	28	24
20 Mont	35	35	35	32
TOTAL	317	310	301	285

## Limitations

**Visit Date attribute.** `-w VisitDates` can not select a visit number unless that visit number appears on a plate containing at least one variable with the `Visit Date` attribute.

**Visit Map.** `-w all` uses `DFX_keys` which contains the keys for each required plate that has been received or registered as missed. Thus **DF\_CTvisits** can only be used for visit numbers that appear on one or more of the required plates defined in the study visit map.

**Site label.** The site label selected with `-l` will only be displayed if `-n` is also included with a value greater than 0. It is not possible to have the site label sized automatically by omitting `-n`.

## See Also

[DF\\_XXkeys](#)

## Name

**DF\_ICcenters** — Verify consistency of the sites database

## Synopsis

```
DF_ICcenters {study} [-e|-w]
```

## Description

**DF\_ICcenters** checks the sites database for common errors and inconsistencies, reporting both errors (problems which will definitely impact the ability to communicate with the sites via DFdiscover) and warnings (problems which may impact communication).

**DF\_ICcenters** should be used to detect errors after **DFcenters** has been defined but before study setup testing is complete. Ideally, **DF\_ICcenters** will list no errors or warnings at the time data collection begins and thereafter.

If **DFcenters** is defined using the Sites view dialog in **DFsetup**, there are already some syntax checks in place that will detect inconsistencies such as out of range values. **DF\_ICcenters** applies additional checks to the sites database that are not applied (or not known) in the Sites view editor. All site definitions that fail these checks are listed as the report output. Each failed check is categorized as either **ERROR** or **WARNING**. The **ERROR** categorization indicates a site definition that will definitely lead to problems in communication. A **WARNING** categorization indicates a site definition that may or may not lead to problems in communication, and hence should be investigated further.

When inconsistencies are detected by **DF\_ICcenters**, each problem is identified with the beginning of the offending record in **DFcenters** followed by the **ERROR** or **WARNING** tag and a brief description of the problem. If multiple problems are detected within the same record, the offending record is only displayed once followed by a list of **ERROR** and/or **WARNING** problems. The exception to this is if problems are detected with the Subject ID entry. **DF\_ICcenters** checks the Subject ID field separately from other fields in **DFcenters**. Hence it is possible for the offending record to appear more than once in the output if it has problems with both the Subject ID and other fields.

**DF\_ICcenters** can be executed from the **DFexplore** Reports View, by using **Verify** available in the Sites view dialog in **DFsetup**, or by using **File > Verify All** also from **DFsetup**.

## Errors

**DF\_ICcenters** currently detects the following **ERROR** conditions:

- Site ID which is not unique and appears in two or more records,
- Records which have fewer than the required minimum of 11 fields,
- Records in which the site ID is missing or blank,
- The site ID is outside the legal range of 0 to 21460 inclusive,
- Records in which the **Contact Person** is missing or blank, or is composed entirely of space/tab characters,
- Records in which the **Affiliation** is missing or blank, or composed entirely of space/tab characters,
- Records in which the **Primary Fax** field contains invalid characters, comma-delimited fax and/or email entries, and is composed entirely of space/tab characters,
- A subject ID range in which the minimum endpoint is greater than the maximum endpoint,
- A subject ID range which overlaps another subject ID range,

- A subject ID range which includes out of range endpoints,
- The `Error Monitor` is defined more than once,
- The `Error Monitor` is not defined at all.

## Warnings

`DF_ICcenters` currently checks for the following `Warning` conditions:

- The `Primary Fax` field includes one or more email addresses but the `Reply To` field does not contain an email address.
- Records which have no subject ID range(s) defined.

## Options

- e Report errors
- w Report warnings

## Name

**DF\_ICimages** — Verify image references against data records

## Synopsis

```
DF_ICimages {study} [-x]
```

## Description

**DF\_ICimages** checks that the CRF image name contained in each database record can be found in the study pages directory (where all documents are stored), and conversely, that each file in the study pages directory is referenced by exactly one database record.

Each database record contains the unique CRF image name corresponding to the submitted CRF page for that data record. **DF\_ICimages** retrieves these image names from all data records and also retrieves the names of all documents stored in the study pages directory. It then compares these 2 lists looking for any cases where there is no one-to-one correspondence between the lists. The result is that this report identifies any database records for which the corresponding image is missing, and any images stored in the study pages directory for which there is no corresponding database record.

**DF\_ICimages** processes the data records first and thereafter processes the CRF images. If the database is receiving new documents at the time that the report is run, it is possible that the results of these two sequential steps will not accurately reflect the database status. To account for this, you may chose to use `-x`.

Without `-x`, the database remains read-write for the duration of the report, and **DF\_ICimages** can erroneously report an inconsistency where there is none. This can occur when new documents are being received at the instant that **DF\_ICimages** is running and the timing of the new pages and data interleaves the execution of **DF\_ICimages**. You may hence choose to ignore inconsistencies that deal with documents received, say today or this week, or run the report with `-x` which will guarantee consistency of results.

With `-x`, **DF\_ICimages** will attempt to place the database server into a read-only state for the duration of the report. If successful, other tools that modify the database, such as **DFexplore** and incoming document processing, will not work until read-write access is restored by **DF\_ICimages**.

## Options

**-x** Request read-only access to the database. This prevents other DFdiscover processes from modifying the database while the report is running, ensuring that the results will not be affected by any new documents that may arrive.

## Name

**DF\_ICkeys** — Check key fields (id,visit,plate) and subject initials

## Synopsis

```
DF_ICkeys {study} [-P #, #-#] [-K IVP] [-R status] [-V vsp] [-I #] [-L sc]
```

## Description

**DF\_ICkeys** checks each id for consistency with the sites database, each visit with the visit map, each plate with the list of legal plates, and optionally checks the consistency of subject initials across all database records within subject.

**DF\_ICkeys** exports the keys for each database record and checks them as follows:

- Ids are counted as errors if they do not appear in the source (i.e., study schema, sites database, or both) specified for legal range definitions. When `-L c` is specified, all subject IDs specified in the sites database are considered legal for all plates defined for the study. With `-L s`, only the study schema is used, where different legal subject IDs may be specified for each plate. If no legal ID specifications are defined for a plate, no legal ID checking is performed on that plate. With `-L sc`, both schema and sites database are consulted, but if a legal range is defined for a plate in the study schema, that definition takes precedence and the sites database is ignored. This allows users to set specific legal criteria at the plate level that override the general specifications in the sites database. This might be useful for side studies where certain CRF forms are only completed at those clinical sites participating in the side study.
- Visit/sequence numbers are counted as errors if they are inconsistent with one or more specifications. The specifications to use can be chosen, using `-V vsp` from the visit map, schema, and/or page map. When the visit map is used, the legal visit/sequence numbers for each plate are those visits for which the plate is specified as required, optional or a missed visit notification plate. When the schema is used, the legal values are those defined in field 6 of each record (this is always the visit/sequence number in a DFdiscover database). When the page map is used, the legal values are those visits for which a visit and plate pair appears in the file. When more than one file is selected, the priority order is: visit map first, then schema, and then page map. If there are no legal visit/sequence numbers specified for a plate, visit checking is disabled for that plate.
- Plate numbers are counted as errors if they differ from the plate data file that contains them (this would be very rare and likely a serious error).
- When `-I` is used to check subject initials, the initials read from the first record (the lowest plate number) read for each subject are saved and compared with the initials on all subsequent records for the same id. If the lowest plate number should not be used as the source of *correct* subject initials, `-P` can be used to alter the plate selection order so that another plate number will be used as the source.

By default, **DF\_ICkeys** checks all 3 keys (id,visit and plate) in records of all statuses (primary, secondary and missed) in all study plates, except for plate 501. It does not check initials by default because there is no default field in which subject initials can always be found.

A subset of the defined plates can be checked using `-P #, #-#`. This option can also be used to change the order in which plates are checked. This is important when used in conjunction with `-I #` because that option uses the first selected plate as the source of the correct subject initials. Queries can be checked for consistency of keys by including plate 511 in the list of plates in `-P #, #-#`, but subject initials will never be checked on that plate.

By default, each of the id, visit, and plate keys is checked. However, by using `-K`, a subset of the keys can be selected by including combinations (with no intervening spaces or commas) of the letters `I` for id, `V` for visit, and/or `P` for plate, in the option string.

`-V` is used to select the source(s) of legal visit/sequence numbers for each plate to use. Any combination (with no intervening spaces or commas) of the letters `v`, `s`, and `p` may be used but they must be specified as a single string. The letter `v` specifies

that the legal visit/sequence numbers for each plate can be determined from the study visitmap. Similarly, *s* indicates that the legal range from the study schema should be used, and *p* indicates that the entries in the study page map should be used. When more than one source is used, the combination of sources is used with precedence given first to visit map, then schema and then page map, when a conflict occurs.

By default, records of all statuses are selected. Use `-R` to select only records of specified statuses. Any legal status keyword from the list: `final`, `incomplete`, `pending`, `missed`, `all` may be used. Multiple keywords must be separated by spaces and the entire string enclosed in `"`.

**DF\_ICkeys** creates a DRF named <sup>1</sup> which lists the records, if any, that had key inconsistencies. This file can be used in **DFexplore** Data View to retrieve and review the problem records. The specific error (id, visit, plate or initials) is written to the 5<sup>th</sup> field of each record in .

## Note

**DF\_ICkeys** reports the number of records with errors for each plate and also reports the number of errors in ids, visit numbers, plates and initials. Some plates may have more than one type of error, and as a result, the number of plates with errors may be less than the sum of the the different types of errors.

## Options

- P #, #-#** Select the plates to be checked (default=all). This option can also be used to change the order of checked plates (this is useful in conjunction with `-I #`).
- R status** Select only records with one of the specified status codes. Valid codes are from the list: `final`, `incomplete`, `pending`, `missed`, `all`
- K IVP** Select the key fields to be checked (default=all), from the list: `I=id`, `V=visit`, `P=plate`
- V vsp** Determine the legal visit/sequence numbers for each plate using the specifications in the:
  - `v`, visit map
  - `s`, legal ranges from schema
  - `p`, page map
  - or a combination thereof.
- I #** In addition to the selected key fields, also check the subject initials that can be found in field # on selected plates.
- L sc** Determine the source for legal subject IDs from:
  - `s`, study schema
  - `c`, sites database
  - `sc`, both schema and sites database

## Examples

### Example 2.12. Check key fields and initials (found in field 8) on plates 1-3

```
-P 1~3 -I 8
DF_ICkeys: Check Key Fields (ID,SEQ,PLT,Initials) DFstudy 7. Apr 22,2017 23:44
Plate 001:      26 errors in 1664 records (26 Initials)
```

<sup>1</sup>This file is overwritten each time that **DF\_ICkeys** is executed.

```

Plate 002:    60 errors in 1675 records (60 Initials)
Plate 003:    44 errors in 1333 records (1 Seqs, 43 Initials)

    Total:    130 errors in 4672 records
Use DF_ICkeys.drf to review problem records.
    
```

### Example 2.13. Check key fields and initials on final and missed records for plates 1-10

```

-R "final missed" -P 1~10
DF_ICkeys: Check Key Fields (ID,SEQ,PLT,Initials) DFstudy 7. Apr 22,2017 23:48

Plate 001:    0 errors in 1143 records
Plate 002:    0 errors in 1106 records
Plate 003:    1 errors in 1139 records (1 Seqs)
Plate 004:    0 errors in 1092 records
Plate 005:    0 errors in 1096 records
Plate 006:    0 errors in 1147 records
Plate 007:    0 errors in 1143 records
Plate 008:    0 errors in 4242 records
Plate 009:    0 errors in 1123 records
Plate 010:    0 errors in 1079 records

    Total:    1 errors in 14310 records
Use DF_ICkeys.drf to review problem records.
    
```

### Example 2.14. Check initials but not the keys on all primary records of plates 11-20

Notice the use of `-K ""` to disable checking on the key fields.

The header line of the output is a standard line and does not reflect the subset of keys selected for the current run.

```

-I 8 -K "" -R "primary" -P 11~20
DF_ICkeys: Check Key Fields (ID,SEQ,PLT,Initials) DFstudy 7. Apr 22,2017 23:51

Plate 011:    0 errors in 1101 records
Plate 012:    4 errors in 1101 records (4 Initials)
Plate 013:    14 errors in 1063 records (14 Initials)
Plate 014:    30 errors in 1974 records (30 Initials)
Plate 015:    19 errors in 914 records (19 Initials)
Plate 016:    8 errors in 926 records (8 Initials)
Plate 017:    11 errors in 926 records (11 Initials)
Plate 018:    10 errors in 923 records (10 Initials)
Plate 019:    16 errors in 921 records (16 Initials)
Plate 020:    1 errors in 57 records (1 Initials)

    Total:    113 errors in 9906 records
Use DF_ICkeys.drf to review problem records.
    
```

### Example 2.15. Check key fields on plates 1-2 using only the study schema as the source for legal subject IDs.



```
-P 1~2 -L s
DF_ICkeys: Check Key Fields (ID,SEQ,PLT,Initials) DFstudy 253. Feb 11,2017 15:07
Plate 001:    30 errors in    53 records (30 IDs)
Plate 002:    11 errors in    26 records (11 IDs)

    Total:    41 errors in    79 records
Use DF_ICkeys.drf to review problem records.
```

## Limitations

It is not possible to compare subject initials across plates where the subject initials variable is not the same variable number on each checked plate.

## Name

**DF\_ICqcs** — Check queries for consistency with data records

## Synopsis

```
DF_ICqcs {study} [ [-r] ]
```

## Description

**DF\_ICqcs** checks the Query database for free-floating queries (i.e. having key fields that do not match an existing data record), duplicate queries (i.e. more than one query attached to the same data field), and unresolved queries bound to final database records.

**DF\_ICqcs** only checks field level queries. It does not check code 21 (missing plate) or code 22 (overdue visit) queries created by **DF\_QCupdate** or code 23 (missing plate) queries created by edit check function `dfaddqc`. Code 21 and 22 queries are removed when the missing plate is entered into the database, or by **DF\_QCupdate** if the plate or visit is no longer missing or overdue, or by the database server when it shuts down, if the subject ID to which the query refers is no longer present in the database. Code 23 queries are only removed when the missing plate is entered into the database. These queries are not touched by **DF\_QCupdate** or by database server shutdown. It would be legal for example, for an edit check to create a code 23 query for a subject that did not exist in the database but should exist according to the study protocol (e.g. when subject IDs must be used in sequence and a gap is found in the subject IDs).

In addition to reporting the number of query inconsistencies found, **DF\_ICqcs** creates a separate output file, containing the problem queries, for each type of query inconsistency found. Each of these files has the current date and time as a common suffix and a prefix matching the type of inconsistency:

- `qcfree` for free-floating queries
- `qcdup` for duplicate queries
- `qcinc` for unresolved queries on final records

The `-r` option can be used to repair 2 of these inconsistencies. When this option is used free-floating queries are deleted and unresolved queries on final records are changed to "resolved, corrected" with the current user and timestamp inserted into the query's resolution field. No attempt is made to resolve duplicate queries.

Duplicate queries can be resolved by using the `qcdup` output file, changing the status field to the value 7 (delete) on the queries you wish to delete, removing the queries that you wish to keep, and then importing the records to be deleted using **DFimport.rpc**.

## Options

- r Repair inconsistencies that are found. The repairs are made by deleting free-floating queries and marking as resolved any unresolved queries found on final records.

## Limitations

This report only checks field-level queries (code 21-23 are ignored), and the repair option is not able to correct duplicate queries.

## Name

**DF\_ICrecords** — Check data records for structural inconsistencies

## Synopsis

```
DF_ICrecords {study} [-p #, #-#] [-r status] [-d] [-s]
```

## Description

**DF\_ICrecords** verifies the structural integrity of data records for all or specified plates of the database, including any pending records. Specifically, **DF\_ICrecords** checks each record for:

1. the correct number of fields as specified by `DFschema`,
2. the correct study number,
3. the correct plate number,
4. a properly formatted creation date/time field, and
5. a properly formatted modification date/time field.

This is not an exhaustive list but has been shown to detect structural problems in database records where they exist. The utility program **DFcmpSchema** should be used to exhaustively check individual data values against the study schema.

**DF\_ICrecords** also checks for:

1. duplicate primary records (more than one primary record with the same keys), and
2. orphaned secondary records (secondary record exists but no primary record can be found with the same keys).

A DRF containing the key fields for each data record that fails this structural consistency check can be created by including `-d`. The DRF is named <sup>1</sup>. This DRF can be subsequently used in **DFexplore** Data View to retrieve and revalidate problem records. Note however that **DFexplore** will only be able to locate the records referenced by the DRF if the record keys are still usable despite the structural inconsistency.

For greater detail than is offered by `-d`, data files with inconsistencies can be exported into 2 files by including `-s`. Each data file that has an inconsistency is exported so that consistent records are written to `plt###.ok` and inconsistent records are written to `plt###.bad`. These files are created in the unique directory `$WORKING_DIR/split.$$` (where `$$` is the program process id). If necessary, the records in `plt###.bad` can be corrected via a manual process and then merged back into the database with the **DFimport.rpc** program.

**DF\_ICrecords** can not check for orphaned secondary records unless records of all statuses are exported (which is the default).

### Status keywords

If `-r status` is used, be careful not to include the Query database `-p #, #-#`. The available status keywords are intended for data record statuses and are not equivalent to query statuses.

By default **DF\_ICrecords** includes records with status `pending`. It is not possible to check pending records separately, but they can be excluded by specifying status `primary`.

<sup>1</sup>This file is overwritten each time that **DF\_ICrecords** is executed.

## Options

- p #, #-#** Select plates to be checked (default=all). The Query Database can be checked by including 511 in the list of plate numbers.
- r status** Select records with the specified status keyword(s) from the list, `final`, `incomplete`, `pending`, `missed`, `all` (default=all).
- d** Create a DRF for any inconsistent records.
- s** Export any data files containing inconsistent records into 2 files: `plt###.ok`, and `plt###.bad`.

## Examples

### Example 2.16. Check only primary records on plates 1-5

```

-r primary -p 1-5
DF_ICrecords: Check Data Records.                               DFstudy 7. Apr 23,2017 21:04

plt001.dat (NF= 28)      1179 Records:      1179 Ok          0 Bad
plt002.dat (NF= 74)      1149 Records:      1149 Ok          0 Bad
plt003.dat (NF= 53)      1151 Records:      1151 Ok          0 Bad
plt004.dat (NF= 40)      1150 Records:      1150 Ok          0 Bad
plt005.dat (NF= 72)      1147 Records:      1147 Ok          0 Bad
    
```

### Example 2.17. Check queries and split any inconsistent records

```

-p 511 -s
DF_ICrecords: Check Data Records.                               DFstudy 7. Apr 23,2017 21:07

DFqc.dat   (NF= 22)      10466 Records:     10466 Ok          0 Bad
    
```

## Name

**DF\_ICschema** — Check database schema for common errors and inconsistencies

## Synopsis

```
DF_ICschema {study} [-P #, #-#] [-a] [-e] [-w] [-SAS #]
```

## Description

**DF\_ICschema** checks the database schema for common errors and inconsistencies, reporting both errors (schema definitions that will definitely lead to problems) and warnings (schema definitions that may lead to problems).

**DF\_ICschema** should be used as a companion to **DFsetup** during study schema development. While **DFsetup** attempts to identify errors as they are made, it cannot catch everything; thus it is important to run **DF\_ICschema** after the setup is complete. It may also be useful to run **DF\_ICschema** periodically during a study setup, for example after setup has been completed for each plate. Ideally, **DF\_ICschema** should report no errors or warnings before the study is started, and thereafter.

All schema definitions that fail **DF\_ICschema** checks are listed in the report output. Each failed check is categorized as either `Error` or `Warning`. An `Error` categorization indicates a definition that will definitely lead to problems. A `Warning` categorization indicates a definition that may or may not lead to problems, and hence should be investigated further.

## Errors

**DF\_ICschema** currently detects the following `Error` conditions:

- Key fields that have not been defined as essential fields,
- Key field formats that contain non-numeric values (i.e. any character other than `ns` or fixed digits 0-9),
- Key fields that do not use a numeric field type (number, choice or check),
- Key fields defined as type choice or check, in which the codes are not unique and legal,
- Key field formats that are longer than 5 characters for visit/sequence numbers and 15 characters for subject ID fields
- Key field store lengths that are not between 1-5 for visit/sequence numbers and 1-15 for subject ID fields,
- Malformed or illegal range specifications,
- If legal values are specified for key field definitions, they must be between 0-65535 for visit/sequence fields and 0-281474976710655 for subject ID fields,
- The `$(ids)` meta-word can be used to indicate that the legal values come from the sites database. This meta-word can only be used to define legal values for the ID field, and can not be combined with any other legal range specification.
- Fields that define a store or display length of 0,
- Choice codes that are repeated within a field definition,
- Failure to define a code for no box checked, in check and choice fields,
- Variable descriptions that are missing or not unique within a plate,

- Total of all store values on a plate that is greater than 16384 characters, the maximum allowed for a single data record. This assumes that all of the data is ASCII characters. If the data contains UNICODE characters, the limit is reduced to 4096.

## Warnings

**DF\_ICschema** currently checks for the following `warning` conditions:

- Inconsistent definition of ID fields across plates, i.e. differences in store length, format and/or legal range.
- Fields that define a display length greater than the store length,
- Field names that are not also valid SAS® variable names
- Variable names that exceed 8 characters in length if SAS® version 6 has been specified,
- Fields that define a format length less than the store length
- Variable descriptions that are longer than 40 characters, if 40 character descriptions are enabled in **DFsetup** preferences; otherwise, descriptions that are longer than 25 characters.

## Options

- P #, #-# Select plates to be checked (default=all).
- a Check field alias (instead of name) for SAS® compatibility
- e Report errors.
- w Report warnings.
- SAS # Select the version of SAS® to be used in applying variable naming rules. The default is SAS® version 7.

## Name

**DF\_ICvisitdates** — List problems detected by the last execution of **DF\_XXkeys**

## Synopsis

```
DF_ICvisitdates {study}
```

## Description

**DF\_ICvisitdates** displays any errors detected during the last execution of **DF\_XXkeys** including:

- visit date inconsistencies (i.e. different visit dates for the same visit),
- illegal visit dates
- visit dates which appear to be out of order, and
- duplicate primary keys.

The source for the report output includes the three files:

- DRF for visit date inconsistencies
- DRF for illegal visit dates
- DRF for duplicate primary records

that were created by **DF\_XXkeys**. **DF\_ICvisitdates** does not create these files itself - it simply formats their contents into a readable report. The check for dates that may be out of order is performed by reading the intermediate file `DFX_visit.dates`, which is also created by **DF\_XXkeys**.

## Options

None.

## See Also

[DF\\_XXkeys](#)

## Name

**DF\_ICvisitmap** — Report inconsistencies and errors in the study visit map

## Synopsis

DF\_ICvisitmap {study}

## Description

**DF\_ICvisitmap** detects syntactic and semantic errors in the visit map, and checks for adherence to DFdiscover visit map rules.

It also checks the definition of all dates defined in the study schema, checks the syntax and validity of page map entries, and checks any of the conditional maps (cycle, visit, plate and termination) that have been defined.

The following checks are performed on the visit map:

1. a screening cycle, if used, precedes all other cycles, and is defined using cycle number 0 and cycle type S,
2. screening cycle contains only visit types X (screening) and O (optional),
3. screening cycle may contain scheduled type X (screening) visits, but the start of the cycle can not be scheduled,
4. an end cycle, if used, is the last cycle in the visit map and is defined using the next sequential cycle number and cycle type E,
5. an end cycle, may not be scheduled and may not contain any scheduled visits, the only visit types allowed are: A (abort), O (optional), and R (required on termination of all follow-up),
6. in-study cycles (i.e. all cycles defined between the screening and end cycles) are numbered sequentially starting from 1,
7. in-study cycles must be one of the following types: R (required), O (optional), or C (conditional),
8. in-study cycle types R and C must have a legal scheduling method (one of: S,B,R,C,N,visit#) that indicates when the study is expected to begin,
9. in-study cycles consisting of only 1 visit, must be defined using visit type B,
10. in-study cycles consisting of 2 or more visits, must have both a baseline and termination visit,
11. the type F (final) visit can only be used in the last in-study cycle,
12. each visit specification contains a single visit number or a valid number list,
13. visit numbers fall within the legal range 0-65535, inclusive,
14. each visit number is unique,
15. each visit is a valid DFdiscover visit type from the list X,P,B,S,O,T,R,r,E,F,A, and W,
16. each visit label is unique, uses only valid visit number substitutions, and is no longer than allowed (17 or 32 characters) depending on study setup preference setting,
17. for those visit types (P,B,S,T,F,A,E,W) which require a visit date, a visit date is specified in the visit map and the specified visit date plate is on one of the required plates for the visit. If the VisitDate plate is not in the list of required plates for the



visit, a warning will be issued. This is acceptable as it is possible that the VisitDate field may be defined on a conditional plate,

18the field identified as the visit date is defined in the study schema using the Visit Date attribute,

19each required plate is listed only once for each visit. It is possible that the required plate list is empty as all plates for a visit may be optional with different ones only becoming required if some condition is met. Thus, **DF\_ICvisitmap** only produces a warning if the plate list is empty,

20due day for pre-baseline visits is  $\leq 0$ ,

21pre-baseline visits precede the baseline visit in each cycle,

22due day must be 0 for all baseline visits,

23baseline visits precede scheduled visits within each cycle,

24due day for each scheduled visit is  $\geq$  due day of any preceding scheduled visits,

25only one type T termination visit appears in each cycle,

26only one type W termination window appears in each cycle,

27if both T and W terminations are defined in a cycle, the type T termination visit must precede the type W termination window,

28only one final visit (type F) appears in the visit map, and it occurs, if used, as the first visit in the last in-study cycle,

29a plate which signals a missed visit is not defined as a required plate,

30cycles scheduled using scheduling method S (from start of first cycle) have a start day which follows the start of preceding cycles,

31termination window syntax, and

32required plates must not signal early termination (except perhaps for early termination (type E) and abort (type A) visits, where defining them this way is acceptable but unnecessary).

The following checks are performed on the conditional maps, if used, for: conditional cycles, visits, plates and termination:

1. the statement that defines the condition has the correct number of fields (5), and references a valid data field to be tested (i.e. the specified visits exist in the visit map, the specified plate exists in the study schema, and the specified field exists on the specified plate),
2. only 1 IF statement is defined for each condition,
3. for the conditional termination map the condition begins with termination type E (early termination) or A (abort all follow-up), and for all other termination maps the condition begins with IF,
4. the conditional termination map consists of condition statements only (termination occurring at the visit that triggers the condition), but the conditional cycle, visit and plate maps also contain one or more action statements following each condition statement,
5. each action statement has 2 fields
6. each action statement begins with either a +, - or ~ sign, to indicate that, when the condition is true, the list of cycles, visits or plates is required, excluded or optional, respectively,

7. for the conditional cycle map the 2nd field of the action statement is a valid list of cycles (i.e. cycles defined in the visit map),
8. for the conditional visit map the 2nd field of the action statement is a valid list of visits (i.e. visits defined in the visit map),
9. for the conditional plate map the 1st field of the action statement contains a valid list of visits (i.e. visits defined in the visit map), and 2nd field contains a valid list of plates (i.e. plates defined in the study schema)

The following checks are performed on the page map, if used.

1. visit acronyms will fit on the subject status list of DFdiscover Query Reports, i.e. the length of visit acronyms plus the length of the longest study VisitDate format is not more than 18 characters,
2. visit acronyms use only visit number substitutions (plate number and data field substitutions are not allowed),
3. plate-sequence labels use valid substitutions (for visit number, plate number or a specified data field),
4. plate-sequence labels will fit on DFdiscover query reports, i.e. are not longer than 25 or 40 characters, depending on which study setup preference has been specified in the study schema,
5. visit acronym specifications reference valid visit numbers (i.e. visits specified in the study visit map),
6. plate-sequence specifications reference valid plate and visit numbers (i.e. visits specified in the study visit map and plates defined in the study schema)

## Options

None.

## Examples

### Example 2.18. Report output for an incorrectly modified version of study 245

```
DF_ICvisitmap: Check Visit Map Specifications.      DFstudy 245. Jan 18,2018 15:24

Study Schema          = /Users/Shared/studies/s245/lib/DFschema
Visit Map             = /Users/Shared/studies/s245/lib/DFvisit_map
Conditional Cycle Map = /Users/Shared/studies/s245/lib/DFccycle_map
Conditional Visit Map = /Users/Shared/studies/s245/lib/DFcvisit_map
Conditional Plate Map = /Users/Shared/studies/s245/lib/DFcplate_map
Conditional Termination Map = /Users/Shared/studies/s245/lib/DFcterm_map
Page Map              = /Users/Shared/studies/s245/lib/DFpage_map

Checking All Date Fields in Study Schema

Checking VisitDate Fields in Study Schema

Checking Visit Map

ERROR at line 7: 93|X|Screen 3|1|10|14|0|1|11|12
      screening visits are only allowed in the screening cycle

Checking Conditional Cycle Map

Checking Conditional Visit Map
```

```
ERROR at line 4: |101-100+value
    syntax error, expected: [+~]|visits(s)

Checking Conditional Plate Map

Checking Conditional Termination Map

Checking Page Map

ERROR at line 10: S|1001-1003,1101-1103|Visit ##{S.1.2}.%{S.3.2}
    label is 12 characters long, the maximum = 10
    example: Visit #10.01
```

## See Also

[DF\\_ICschema](#)  
[DF\\_QCupdate](#)  
[DF\\_SSvisitmap](#)

## Name

**DF\_PTcrfs** — Display available CRF information for subjects

## Synopsis

```
DF_PTcrfs {study} [-c #, #-#] [-i #, #-#] [-p] [-x] [-l #]
```

## Description

**DF\_PTcrfs** lists all assessments received for each subject with the date of the assessment and the status of all pages of the CRF received for that visit, including both required and optional pages.

Each row of the output displays the subject ID, the visit number, the visit date if known, and a list of plate numbers received for that subject ID and visit. Each plate number has an optional suffix that identifies the highest priority problem on that plate, if any. The code and meaning of each suffix is shown in the figure. Those in the first column are not mutually exclusive and are listed in priority order. For example, a plate which has both an unresolved Q&A note and an unresolved internal query will have the suffix ? because the Q&A note takes precedence. A plate number that has no suffix is considered to be final.

### Figure 2.1. Legend of suffix codes and their meaning

QUERY CODES (ordered by priority):	OTHER CODES:
r = outstanding refax request	- = missing page
? = unresolved Q&A note	x = registered as missed
i = unresolved internal query	* = error status (missing keys)
+ = incomplete but no unresolved queries	p = pending status
	NA = visit date is coded Not Available

The - suffix used to identify missing plates is determined from the Query database where missing plate queries are inserted by **DF\_QCupdate**. **DF\_QCupdate** must be executed to update the Query database before running **DF\_PTcrfs** if the status of missing plates is likely to have changed, generally because new records have been validated since **DF\_QCupdate** was last run, or the visit map was changed.

## Options

- c #, #-#** Select only subjects from specified sites. By default, subjects from all sites, including the error monitor site, appear in the output.
- i #, #-#** Select only subjects with specified IDs. By default, all subjects from all sites, including the error monitor site, appear in the output.
- p** Execute **DF\_XXkeys**, but only list plates with problems. With this option, plates that would otherwise have no suffix are not displayed.
- l #** Set maximum length of output line to #chars (default # = 80)

## Examples

### Example 2.19. List CRF information for all subjects at site 99

```
-c 99
DF_PTcrfs: CRFs Received at Each Visit.           DFstudy 254. Dec 09,2014 14:39
```

Complete CRF Log by Subject : Acceptance Test Kit

QRY CODES (ordered by priority):

r = outstanding refax request  
 ? = unresolved Q&A note  
 i = unresolved internal query  
 + = incomplete but no unresolved queries

OTHER CODES:

- = missing page  
 x = registered as missed  
 \* = error status (missing keys)  
 p = pending status  
 NA = visit date is coded Not Available

SUBJECT	VISIT	DATE	CRF PLATES
99001	0	03/03/97	1
99001	1	06/03/97	2 3 4
99001	21	06/04/97	5
99001	22	06/05/97	5
99001	23	06/06/97	5
99001	24	06/07/97	5
99001	30	06/07/97	6- 7
99001	51		8
99001	201		11
99001	1011		9
99002	0	04/01/97	1
99002	1	08/01/97	2 3 4
99002	21	08/02/97	5
99002	22	09/03/97	5
99002	23	10/04/97	5
99002	24	07/05/97	5
99002	30	07/05/97	6 7
99002	40	12/06/97	10
99002	51		8
99002	1011		9p
99003	0	07/01/97	1
99004	0	01/06/97	1
99004	1	03/06/97	2 3 4
99004	21	04/07/97	5
99004	22	05/08/97	5
99004	23	06/09/97	5
99004	24	06/10/97	5
99004	30	06/10/97	6 7
99004	201		11
99006	201		11
99007	0	23/04/97	1
99007	1	25/04/97	2p 3 4x

**See Also**

[DF\\_XXkeys](#)

## Name

**DF\_PTlist** — List subject data grouped by subject ID and visit number

## Synopsis

```
DF_PTlist {study} [-u] { { [-X plate field~list] | [-x plate field format] | [-F format plate field~list] |
[-R plate field~list] } [ [-S selectcondition] | [-s selectcondition] ] [-e] ...} [ [-H variable_names] | [-U] | [-
G] ] [-l #, #-#] [-c #, #-#] [-v #, #-#] [ [-b lines #] | [-b cases #] ] [-d] [-t] [-l] [-f file] [-p]
```

## Description

**DF\_PTlist** is a simple subject-listing program which displays subject data in a spreadsheet-like format. The columns are selected variables (e.g. age sex, diagnosis, etc.), and the rows are individual subject ID and visit number combinations on which the data was collected. **DF\_PTlist** is simple to use, but also limited. It is not intended to be a general-purpose report generator and will not replace your need for more sophisticated reporting software.

**DF\_PTlist** relies heavily on the ability to specify field and plate numbers. As a result it is useful to have a current listing from **DF\_SSschema** or **DF\_SSvars** available when constructing the option list.

The diversity of uses for **DF\_PTlist** is best learned by following the examples.

## Options

<b>-u</b>	Display program usage, including a few examples, and exit
<b>-X <i>plate field~list</i></b>	List fields with default formatting
<b>-x <i>plate field format</i></b>	List fields with specified formatting
<b>-F <i>format plate field~list</i></b>	List fields using specified format string
<b>-R <i>plate field~list</i></b>	Repeat/normalize fields in previous -x -X record
<b>-S <i>selectcondition-s</i> <i>selectcondition</i></b>	Select subjects (with <i>-s</i> ) or records (with <i>-S</i> ) meeting these criteria
<b>-e</b>	End/remove last <i>-s</i>  - <i>S</i> selection criteria
<b>-H <i>variable_names</i></b>	Variable column headers for previous retrieval
<b>-U</b>	Use variable alias from DFschema. This can also be specified using -H DFalias
<b>-G</b>	Use variable name from DFschema (default). This can also be specified using -H DFname
<b>-l #, #-#</b>	Select only specified subjects. By default, all subjects from all sites, including the error monitor site, appear in the output.
<b>-c #, #-#</b>	Select only subjects from specified sites. By default, subjects from all sites, including the error monitor site, appear in the output.
<b>-v #, #-#</b>	Select only specified visits
<b>-b <i>lines #</i> -b <i>cases #</i></b>	Insert page breaks into output after specified lines or cases (subjects)

<b>-d</b>	Data only. Writes   delimited ASCII records
<b>-t</b>	Translate codes to value labels, where defined
<b>-l</b>	Include missed records (values displayed using x)
<b>-f file</b>	Read options from the specified file. When options are stored in a file and read in this manner, the full path name to <i>file</i> must be specified, unless <b>DF_PTlist</b> is being executed from a UNIX command-line, in which relative pathnames are taken relative to the directory that the report is invoked from.
<b>-p</b>	Display status and validation level for each plate (e.g. 1f6 = plate 1, final, level 6)

## Examples

These examples are based upon study 254, the DFdiscover Acceptance Test Study. The number of records in the database has been intentionally limited so that the examples can demonstrate the input with very little output.

### Example 2.20. List, with default formatting, fields 9 and 21-25 from plate 1, and fields 10 and 12 from plate 5

```
-x 1 9 21~25 -x 5 10 12
DF_PTlist: Subject Data Listings.                DFstudy 254. Nov 11,2017 08:10
```

SUBJECT	VISIT	AGE	S2SBP1	S2DBP1	S2SBP2	S2DBP2	S2SBP3	FSBP1	FSBP2
99002	0	64	172	098	168	096	166	.	.
99002	21	.	.	.	.	.	.	154	150
99002	22	.	.	.	.	.	.	150	146
99002	23	.	.	.	.	.	.	144	140
99002	24	.	.	.	.	.	.	136	138

### Example 2.21. From plate 1, list fields 8 and 9, right justified in columns that are 10 and 12 characters wide, respectively

```
-x 1 8 %10s 9 %12s
DF_PTlist: Subject Data Listings.                DFstudy 254. Nov 11,2017 08:12
```

SUBJECT	VISIT	PINIT	AGE
99002	0	JAM	64

### Example 2.22. Normalize the 3 blood pressure measurements in fields 21 through 26 on plate 1

```
-x 1 21 22 -R 1 23 24 -R 25 26
DF_PTlist: Subject Data Listings.                DFstudy 254. Nov 11,2017 08:29
```

SUBJECT	VISIT	S2SBP1	S2DBP1
99002	0	166	094
99002	0	168	096
99002	0	172	098

Notice how the default column labeling is taken from the variables requested by `-x`. This can be changed with the addition of `-H new_names` as in:

```
-X 1 21 22 -H SBP DBP -R 1 23 24 -R 25 26
DF_PTlist: Subject Data Listings.                DFstudy 254. Nov 11,2017 08:44
```

SUBJECT	VISIT	SBP	DBP
99002	0	166	094
99002	0	168	096
99002	0	172	098

**Example 2.23.** List cases in which the 5<sup>th</sup> field extracted equals 1, where  $x[i]$  is the  $i^{\text{th}}$  field in the next `-x`, `-X` or `-R`

```
-S "if(x[5]==1)" -X 5 10-14
DF_PTlist: Subject Data Listings.                DFstudy 254. Nov 11,2017 08:52
```

SUBJECT	VISIT	FSBP1	FDBP1	FSBP2	FDBP2	FARM
99002	21	154	092	150	090	1
99002	23	144	090	140	086	1
99002	24	136	082	138	080	1

**Example 2.24.** List field 8 through 10 of plate 1 for subjects 99000-99999

```
-X 1 8-10 -I 99000-99999
DF_PTlist: Subject Data Listings.                DFstudy 254. Nov 11,2017 08:55
```

SUBJECT	VISIT	PINIT	AGE	SEX
99002	0	JAM	64	1

**Example 2.25.** Print a page break and column titles every 50 lines

In this case the option would be a part of a complete option string.

```
-b lines 50
```

**Example 2.26.** Print every subject on a new page

The data for this example are not from the same study 254 as the other examples. There are also many other options specified. The important one in this case is `-b cases 1`.

```
-X 1 3 9 -X 2 3 9 -v 1 3~5 -I 1101 1103 -p -b cases 1
DF_PTlist: Subject Data Listings.                DFstudy 253. Nov 11,2017 08:57
```

SUBJECT	VISIT	STATUS	DFRASTER	edate	DFRASTER	vdate002
1101	1	1f6	9532/0040005	07/08/95	.	.
1101	1	2f6	.	.	9532/0054001	07/09/95
1101	3	1f6	9533/0177004	07/14/95	.	.
1101	4	1f6	9533/0177007	08/10/95	.	.
1101	5	1f6	9540/0098002	10/04/95	.	.

pagebreak



SUBJECT	VISIT	STATUS	DFRASTER	edate	DFRASTER	vdate002
1102	1	1f6	9532/0035007	07/24/95	.	.
1102	1	2f6	9532/0035007	07/24/95	9532/0035001	07/24/95
1102	3	1f6	9534/0119003	07/27/95	.	.
1102	4	1f6	9534/0125007	08/25/95	.	.
1102	5	1f6	9542/0105002	10/20/95	.	.

-x is used to select fields 3 and 9 from plates 1 and 2. This option produces automatic formatting of columns, with each column being just as wide as it needs to be to accommodate the column title and data.

-p requests the plate number, status, and validation level of each record from which data is extracted. This is displayed in the column with the STATUS heading and is a single string of 3 to 5 characters in length, where the first digits up to the first character are the plate number, the single character is a code for the record status from the list: f=final, i=incomplete, p=pending, L=missed, and the trailing digit is the record's validation level.

The column headings SUBJECT and VISIT are fixed by DF\_PTlist, they cannot be changed nor removed. All other titles are the variable names from the study schema.

-v selects only visits 1, 3, 4 and 5; otherwise, all visits for which data values are available would be displayed.

Data values that do not exist in the database because that plate is not present for the subject, are indicated by a single dot. In this example plate 2 only occurs at visit 1 and thus dots appear under the plate 2 data columns for visits 3-5.

### Example 2.27. List fields 9 to 12 from plate 1, and fields 15, 12 and 16 (in that order) from plate 2

```
-X 1 9~12 -X 2 15 12 16
DF_PTlist: Subject Data Listings. DFstudy 254. Nov 11,2017 09:25
```

SUBJECT	VISIT	AGE	SEX	RACE	RACEOTH	HTIN	WTKG	PULSE
99002	0	64	1	1	.	.	.	.
99002	1	.	.	.	.	072.0	.	056

The RACEOTH variable in the first row, and WTKG in the second row, are blank observations (which are different from observations marked with . which indicate that the record containing the observation is missing).

### Example 2.28. List fields 8 and 9 from plate 1, formatted using field widths of 5 and 12 characters respectively

```
-x 1 8 %5s 9 %12s -c 12~14 22 -I 6001 -v 1 51~59 -b lines 50
```

In addition to formatting, this option string requests:

- Subject at sites 12,13,14 and 22 (-c), plus subject 6001 (-I), and visits 1 and 51 to 59 (-v)
- a page break every 50 lines (-b)

When using -x with specified formatting ensure that the specified field widths are wide enough to accommodate the widest value and the column label. Values or labels wider than the specified width are displayed intact resulting in misalignment of subsequent columns. To truncate fields at a desired width, say 5 characters, use a format string of %5.5s.

### Example 2.29. List fields 12 to 16 from plate 5 where field 12 of plate 5 equals 1, and fields 21 and 22 from plate 8

```
-S "if(x[1]==1)" -X 5 12~16 -e -X 8 21 22
```

Fields 12 to 16 map to `x[1]` through `x[5]` for the `-s` case selection coding. Also note that `-e` is necessary to turn off the case selection code so that it is not applied to the selection of fields from the next `-x` specification (i.e. for plate 8). If `-e` was not specified, the selection code would also be applied to `x[1]` of plate 8, that being field 21. The result would be that only cases that met both selection criteria would be printed.

It is legal to have a different case selection specification for each `-x|-x`. For example if field 12 on plate 5 was gender (coded 1=male, 2=female), and fields 21 and 22 on plate 8 were systolic and diastolic blood pressure readings, we could select male subjects with hypertension by specifying:

```
-s "if(x[1]==1)" -x 5 12~16 -s "if(x[1]>140 || x[2]>90)" -x 8 21 22
```

### Example 2.30. List multiple events, e.g. adverse events, for each subject

```
-x 1 9 10 -s "if(x[1]>0)" -x 7 9 %9s 10 %9s 11 %20.15s -R 7 12~14 -R 7 15~17
```

The first `-x` specification gets fields 9 and 10 from plate 1 (e.g. `age` and `sex`). Next suppose that we want to list 3 fields for each adverse event: an AE code, start date, and description. The `-s` specification results in the creation of records if the AE code is greater than zero. The `-x` specification identifies where the first AE is recorded and specifies the output format for each field. AE code and start date will be printed in fields 9 characters wide, and the description will be limited to the first 15 characters, output in a space 20 characters wide. Remember to make fields wide enough to allow some space between fields so that the field columns can be clearly distinguished. The `-R` (repeat) specifications identify where other adverse events are recorded. They will be printed using the formatting given in the preceding `-x`. The preceding `-s` will also be applied to these fields, so that records will only be created if an adverse event exists.

The result will be a list that includes at least age and sex for all subjects, plus a list of adverse events for those subjects that have experienced them. If `-s` was changed to `-S`, the meaning would be different. `-S` indicates case selection criteria (as opposed to record selection), and thus would result in only listing those subjects who had experienced adverse events.

Multiple `-s` and `-S` criteria may be used in the same report. Each criterion remains in effect until the next one is specified. To list only subjects over 50 who have had adverse events, the input string could be modified to:

```
-s "if(x[1]>50)" -x 1 9 10 -s "if(x[1]>0)" -x 7 9 %9s 10 %9s 11 %20.15s -R 7 12~14 -R 7 15~17
```

### Example 2.31. Using `-e` to end previous selection criteria

```
-s "if(x[1]>50 && x[2]==1)" -x 1 9 10 -e -x 2 13~17
```

Since `-s` and `-S` selection criteria remain in effect for subsequent `-x|-x` specifications, a method of ending previous selection criteria is needed when they are not to be applied to the next `-x|-x`.

This is accomplished by `-e`. In this example, age and sex, from fields 9 and 10 of plate 1, are to be listed along with vital signs from fields 13 to 17 of plate 2. Further we want to limit the list to men over 50. The `-s` specification used to select men over 50 must be turned off before the `-x` specification for vital signs. Otherwise, `-s` selection criteria would be applied to the vital signs as well, i.e. only men over 50 who also had plate 2, field 13 > 50 and plate 2, field 14 equal to 1, would be listed. In this example it is unlikely that this would be desired (or that it would produce any output). However in those cases where the same variables are stored on several plates and the same selection criteria are desired for the fields pulled from each plate, having the `-s` selection criteria carry forward is exactly what is needed.

## See Also

[DF\\_SSvars](#)  
[DF\\_SSschema](#)

## Name

**DF\_PTmissing** — Display missing pages and overdue visits

## Synopsis

```
DF_PTmissing {study} [-c #, #-#] [-o] [-m] [-e] [-p]
```

## Description

**DF\_PTmissing** reads the Query database and lists missing plate and overdue visit queries for all subjects, or for subjects from specified study sites.

Missing plates and overdue visits are identified by the most recent execution of **DF\_QCupdate** and are simply reported by **DF\_PTmissing**.

If none of `-o`, `-m` and `-e` are selected, overdue visits and both types of missing plates are listed. Any combination of these 3 problem types may be listed by selecting the appropriate options.

## Options

- c #, #-#** Select only subjects from specified sites. By default, subjects from all sites, including the error monitor site, appear in the output.
- o** Display overdue visits
- m** Display missing plates
- e** Display missing plates added via edit checks
- p** Display missing plates by their `DFpage_map` labels

## Examples

### Example 2.32. Display overdue visits and missing plates for site 1

The default output lists overdue visits by their name as stated in the study visitmap, and missing plates by their plate number.

```
-c 1
SUBJECT ID  VISIT  PROBLEM
           6      1  Overdue Visit:  Initial Assessment
           6      2  Missing Plates: 4 8 9 10 11 12 13 14 15 16
```

### Example 2.33. Display overdue visits and missing plates, by their name, for site 1

This example selects the same subjects as the previous example, except that in this instance `-p` requests plate numbers be replaced by their descriptive labels from `DFpage_map`.

```
-c 1 -p
SUBJECT ID  VISIT  PROBLEM
```

```
6      1  Overdue Visit:  Initial Assessment
6      2  Missing Plates: 4M Page 1 of 6
                               4M Page 5 of 6
                               4M Page 6 of 6
                               Nurse/Diet. Summ.
                               4M FIT QOL Pg 1
                               4M DQOL Pg 1 of 5
                               4M DQOL Pg 2 of 5
                               4M DQOL Pg 3 of 5
                               4M DQOL Pg 4 of 5
                               4M DQOL Pg 5 of 5
```

## Limitations

**DF\_PTmissing** lists the missing plate and overdue visit queries added by the previous execution of **DF\_QCupdate**. It does not automatically execute **DF\_QCupdate**.

## See Also

[DF\\_QCupdate](#)

## Name

**DF\_PTqcs** — Summary report of external queries for each subject

## Synopsis

```
DF_PTqcs {study} {-l #, #-#} [-x] [-a] [ [-t yy/mm/dd] | [-t] | [-t yy/mm/dd-yy/mm/dd] | [-t] ]
```

## Description

**DF\_PTqcs** is identical in function to **DF\_CTqcs** but it groups data by subject id range (required) whereas **DF\_CTqcs** groups data by site.

**DF\_PTqcs** reports a summary of the user-created, external queries added during data validation. queries are selected by their creation date from a specified time period that may include one of:

- from the beginning of the study
- from a specified date
- between two specified dates

The following queries are not included in the report:

- Internal queries,
- Missing page queries (added by **DF\_QCupdate** or edit checks), and
- Overdue visit queries (added by **DF\_QCupdate**).

The summary is in the form of two tabular reports. The first report displays for each subject (row), columns that contain:

### Example 2.34. First Tabular Report

SUBJECT ID#	TOTAL RECORDS	TOTAL QRYS	RATE/100 REC	RESOLVED #	QRYS %	STATUS	TIME TO RESOLUTION	Pending
	①	②	③	④	⑤	Correct	N.A.	Irrel. DAYS

- ① Number of primary records created
- ② Number of external queries created during validation
- ③ QRY rate = number of queries created per 100 primary records created
- ④ Number and percentage of these queries which have been resolved
- ⑤ Number of each resolution type from the list:
  - corrected
  - not available
  - irrelevant
- ⑥ Mean time (in days) from the creation of the queries to their resolution
- ⑦ Number of queries that have replies that are pending review

The second table displays for each subject (row), columns that contain:

## Example 2.35. Second Tabular Report

SUBJECT	AGE (DAYS) NEW/UNSENT QUERIES <sup>①</sup>					AGE (DAYS) SENT/UNRESOLVED QUERIES <sup>②</sup>				
ID#	00-10	11-20	21-30	31-60	> 60	00-10	11-20	21-30	31-60	> 60

- ① Categorical time (in days) from query creation to today's date, for unresolved queries that have not been sent to the sites in a Query Report.
- ② Categorical time (in days) from query creation to today's date, for unresolved queries that have been sent to the clinical sites. Queries with status pending are included in this category.

An alternative report format can be requested with `-a`. This format displays results in a single table that is a hybrid of the two tables described above.

## Figure 2.2. Alternative Report from `-a`

SUBJECT	TOTAL	TOTAL	RATE/	RESOLVED	QRYS	AGE (DAYS) OF UNRESOLVED QRYS				
ID#	RECORDS	QRYS	100 REC	#	%	<=10	11-20	21-30	31-60	>60

**DF\_PTqcs** uses summary file which is created by **DF\_XXtime**. **DF\_PTqcs** will run **DF\_XXtime** first, if necessary, to update unless `-x` is used, in which case the results will reflect the state of the database on the last time that **DF\_XXtime** was run.

## Options

- `-l #, #-#` Include only specified subjects (required)
- `-a` Alternative format shows age of unresolved queries (instead of query status)
- `-t yy/mm/dd-t` Include records and queries created since specified date, inclusive of the specified date
- `-t yy/mm/dd-yy/mm/dd-t` Include records and queries created between specified dates, inclusively

## Limitations

1. This report cannot be used to obtain the status of internal queries or external queries created for missing pages and overdue visits.
2. Mean days from creation to resolution is based on those queries that have been resolved so far and thus may not reflect the true average time if there are many unresolved queries of long duration.

## See Also

[DF\\_XXtime](#)  
[DF\\_CTqcs](#)

## Name

**DF\_PTschedule** — Display subject scheduling information from **DF\_QCreports**

## Synopsis

```
DF_PTschedule {study} [-c #, #-#] [ [-p all] | [-p fup] | [-p qcs] | [-p notdone] | [-p done] ]
[-idfmt #] [-dtfmt fmt]
```

## Description

**DF\_PTschedule** shows the first visit, last completed visit, and next scheduled visit for each subject. It also shows if there are any outstanding problems, i.e. unresolved queries.

This report uses **DF\_QCreports** to produce an internal Query Report listing the status of subjects from the specified sites. The purpose of **DF\_PTschedule** is to provide a simpler mechanism (than **DF\_QCreports**) for obtaining subject status information only. The only additional functionality added by **DF\_PTschedule** that is not offered by **DF\_QCreports** is through the `-p done` option.

## Options

- c #, #-#**      Select subjects from specified sites. By default, subjects from all sites, including the error monitor site, appear in the output.
  
- p all-p fup-p qcs-p notdone-p done**      Select which subjects to include from the following list.
  - `-p all`: all subjects
  - `-p fup`: subjects still under follow-up
  - `-p qcs`: subjects with outstanding queries
  - `-p notdone`: subjects under follow-up or with unresolved queries
  - `-p done`: subjects finished follow-up and no unresolved queries
  
- idfmt #**      Leading zero-pad IDs on output report to the specified number of digits. For example, `-idfmt 6` prints IDs as 6 digits with leading zeros. The default is no zero padding of IDs.
  
- dtfmt *fmt***      Set the output format for dates. Any combination of `dd`, `mm` or `mmm`, `yy` or `yyyy` and delimiters (space, commas, etc.) can be used.

## Examples

### Example 2.36. show visit scheduling info for subjects still under follow-up at sites 10 and 99

```
-c 10 99 -p fup -dtfmt "mmm dd, yyyy"
DF_PTschedule: Subject Visit Scheduling.                      DFstudy 254. Apr 23,2017 22:28

SUBJECT STATUS SUMMARY (* identifies subjects with outstanding data queries)

SUBJECT                      ENTRY VISIT                      LAST FOLLOW-UP                      NEXT FOLLOW-UP
```

10100*	Day 1: JAN 01, 2016	3 Mo.: APR 01, 2016	4 Mo.: MAY 03, 2016
10101*	Day 1: OCT 01, 2016	3 Mo.: JAN 01, 2017	4 Mo.: JAN 31, 2017
10102*	Day 1: JAN 01, 2016	3 Mo.: APR 01, 2016	4 Mo.: MAY 03, 2016
10103*	Day 1: JAN 01, 2015	3 Mo.: unknown	4 Mo.: MAY 03, 2015
10104*	Day 1: NOV 01, 2015	3 Mo.: JAN 01, 2016	4 Mo.: MAR 03, 2016
10105*	Day 1: OCT 01, 2016	3 Mo.: DEC 30, 2016	4 Mo.: JAN 31, 2017
10200*	Day 1: JAN 02, 2016	Day 1: JAN 02, 2016	1 Mo.: FEB 01, 2016
10300*	Day 1: JAN 02, 2016	Day 1: JAN 02, 2016	1 Mo.: FEB 01, 2016
10400*	Day 1: JUN 01, 2016	1 Mo.: JUL 02, 2016	2 Mo.: AUG 01, 2016
10500*	Day 1: JUN 15, 2016	1 Mo.: JUL 15, 2016	2 Mo.: AUG 15, 2016
10600*	Day 1: OCT 01, 2015	4 Mo.: MAR 01, 2016	End: JAN 31, 2016
10601*	Day 1: OCT 08, 2016	Day 1: OCT 08, 2016	1 Mo.: NOV 07, 2016
99002*	Day 1: MAR 06, 2015	Day 1: MAR 06, 2015	1 Mo.: APR 05, 2015

## Limitations

The total space allotted for the visit label and date is 23 characters. Allowing a maximum of 6 characters for the label, 2 characters for ": " between the label and the date, and 2 characters for space between visits, leaves a maximum size for date formats of  $23 - 6 - 2 - 2 = 13$  characters.

## See Also

[DF\\_QCreports](#)



## Name

**DF\_PTunexpected** — List unexpected data records found in the study database

## Synopsis

DF\_PTunexpected {study} [[-i #, #-#] | [-c #, #-#] ] [-v #, #-#] [-p #, #-#]

## Description

This report lists unexpected plates and visits found in the study database by **DF\_QCupdate**. The report output appears as illustrated in the following example.

```
DF_PTunexpected: Unexpected Data by Subject.          DFstudy 2. Jan 25,2016 12:08

 99010 Visit 1019: Phase I: 1-Mon, Extra Visit 9
      Plate(s)   : 24,30-33,35-36
      Problem    : unexpected visit: 18 days post termination

 99011 Visit 1020: Phase I: 2-Mon Visit
      Plate(s)   : 30-34
      Problem    : unexpected plate: excluded by plate condition 24

 99011 Visit 1020: Phase I: 2-Mon Visit
      Plate(s)   : 75
      Problem    : unexpected plate: not required, optional or conditional

 99011 Visit 1021: Phase I: 2-Mon, Extra Visit 1
      Plate(s)   : 20,24,30-36
      Problem    : unexpected visit: excluded by visit condition 5
```

The output for each problem consists of 3 lines. The 1st line identifies the subject ID, visit number and visit label. The 2nd line identifies the plate or plates that were unexpected at that visit. The 3rd line describes the problem as follows:

- **unexpected plate: excluded by plate condition #.** This problem description identifies plates that have been received unexpectedly because they were excluded by one of the conditions specified in the conditional plate map. Use **DF\_SSvisitmap** to get a numbered list of the conditions in each conditional map.
- **unexpected plate: not required, optional or conditional.** This identifies plates that have been received for a visit which are not defined as required or optional in the visit map and have not been triggered by a conditional plate map specification. If such plates are allowed, and perhaps optional, they should be included in the visit map specification for each visit at which they might be used.
- **unexpected plate: a missed visit plate exists for this visit.** A missed visit plate can be defined in the visit map for each visit. Typically a missed visit plate corresponds to a one page CRF form which is submitted by the clinical sites to give notice that a required subject assessment can not be completed because the subject missed the visit, refused the assessment, became ineligible for the assessment, etc. It is thus inconsistent for the site to submit a missed visit form and also to submit one or more plates for the same visit.
- **unexpected visit: # days post termination.** If a termination event has occurred within a cycle we do not expect any visits to be performed within that cycle following the termination date. And if an abort event has occurred we do not expect any visits to be performed after the abort date in any of the study cycles. **DF\_PTvisits** can be used to list all visits completed for a subject and to identify those visits at which a termination event occurred.

- **unexpected visit: excluded by visit condition #.** This problem description identifies visits that have been received unexpectedly because they were excluded by one of the conditions specified in the conditional visit map. Use **DF\_SSvisitmap** to get a numbered list of the conditions in each conditional map.
- **unexpected visit: excluded by cycle condition #.** This problem description identifies visits that have been received unexpectedly because they were excluded by one of the conditions specified in the conditional cycle map. Use **DF\_SSvisitmap** to get a numbered list of the conditions in each conditional map.
- **unexpected visit: aborted in prior cycle.** If an abort event has occurred we do not expect any visits to be performed after the abort date in any of the study cycles. **DF\_PTvisits** can be used to list all visits completed for a subject and to identify those visits at which a termination event occurred.

## Options

If no options are specified **DF\_PTunexpected** lists all of the unexpected records identified by the last execution of **DF\_QCupdate**. The following options can be used to focus the output on selected cases.

- i #, #-# select subjects by ID
- c #, #-# select subjects by site ID
- v #, #-# select problems by visit number
- p #, #-# select problems by plate number

## Limitations

1. The input for **DF\_PTunexpected** comes from DFdiscover retrieval file `DFunexpected.drf` (created by **DF\_QCupdate**). Thus, the output will include unexpected data records identified by the most recent execution of **DF\_QCupdate**, and will not contain any changes that have occurred since then.
2. Cycle, visit and plate requirements are not fixed and may change during the trial. For example, a plate may be unexpected because the data value that makes it required has not yet been entered into the database, or has been entered incorrectly. It is thus necessary to manually review these cases before generating data queries for the clinical sites. The best way to accomplish this is to use **DF\_PTunexpected** to produce a list of the current problems, and then use `DFunexpected.drf` to retrieve and review these cases in **DFexplore** Data View, where data entry errors can be corrected and queries can be created as needed.

## See Also

[DF\\_QCupdate](#)  
[DF\\_PTschedule](#)

## Name

**DF\_PTvisits** — Scheduling and status of all cycles & visits in the visit map

## Synopsis

```
DF_PTvisits {study} [-itime {[[e#] | [l#] | [o#] | [oU]...} [[done] | [notdone]]}] [ [-i #, #-#] | [-c #, #-#] ]
[-v #, #-#] [-vtype {[X] | [P] | [B] | [O] | [S] | [T] | [W] | [F] | [E] | [A] | [R] | [r...]}] [-vtime {[e#] | [l#] | [o#] | [oU]...}]
[-cneed {[all] | [[[unknown] | [required] | [next] | [optional] | [excluded]...]]}] [-cdone {[all] | [[[notdone] | [overdue] | [inprogress]
| [terminate] | [abort]...]]}]
[-vneed {[all] | [[[unknown] | [required] | [next] | [optional] | [excluded]...]]}] [-vdone {[all] | [[[notdone] | [overdue] | [received]
| [terminate] | [abort]...]]}]
[-tag {[[all] | [[[lost] | [mvp] | [etp] | [CT] | [CV] | [overdue]...]]} [[e#] | [l#] | [o#] | [oU]...} [[sday] | [cday]]}] [-fmt fmt] [-pg rows
cols 0/1]
```

## Description

This report shows visit scheduling and status information for subjects and visits meeting the specified selection criteria. The report output appears as illustrated in the following example.

```
DF_PTvisits: Visit status & schedule. DFstudy 245. Mar 20,2016 11:35 PAGE 35
```

```
1064 0:SCREENING (required) -terminated 03/09/07
1064 91 X Screen #1 0 0 rD ~03/08/31
1064 92 X Screen #2 7 0 rT 03/09/07
1064
1064 1:IN-STUDY VISITS (required)
1064 51 P Pre-entry -2 0 rD ~03/09/11
1064 0 B Baseline 0 0 rD 03/09/13
1064 1 O QOL form - - rD 03/09/13 CV1r
1064 61 r Lab Test - - x. CV2x
1064 3 S Month 3 91 5 rL
1064 6 S Month 6 183 5 n* ~04/03/14 DOD=6
1064 9 S Month 9 274 5 r. ~04/06/13
1064 12 T Month 12 365 5 r. ~04/09/12
1064 81 E Early Term - - o.
1064 210 R Clinical Eval 0 0 r. ~04/09/12
1064 211 R Patient Eval 30 0 ?.
1064
1064 2:REPORTS (end)
1064 80 A Death Report - - o.
1064 101 O AE Report #01 - - o.
```

The output for each subject is displayed as a block of records beginning with the subject ID (1064 in the example above).

**Cycle Records.** Each cycle begins with a record showing the cycle number, label, type, conditions that affect cycle requirement (if any), and the cycle termination date (if it has terminated). The termination date may also identify an abort date that precedes the scheduled cycle start date. If the cycle requirement is affected by a conditional cycle map entry the condition number and action is displayed using the following format:

```
CC# (required | optional | excluded)
```

where # is the condition number determined by the order in which the conditions are defined in the conditional cycle map.

**Visit Records.** Each visit starts with the visit number, the single letter visit type acronym, visit label, visit scheduled due day, and the overdue allowance, all of which come from the study visit map. For optional visits, which cannot be scheduled, due day and overdue allowance appear as - (dash). These visit map specifications are followed by visit status and scheduling information in the following order:

1. **visit need.** A single letter tag indicates whether the visit is required:

r = required  
 o = optional  
 x = excluded  
 n = next required visit  
 ? = requirement not yet known

2. **visit status.** A single letter tag indicates whether the visit has been completed:

D = visit done  
 T = visit done and cycle terminates at this visit  
 A = visit done and all cycles terminate at this visit  
 L = visit is recorded as missed  
 . = visit has not been received and is not yet overdue  
 \* = visit has not been received and is overdue

3. **visit date.** If the visit has been received and the visit date is known the visit date is shown. If the visit has not been received, or has been received but the visit date is unknown, the scheduled visit date is shown with a ~ (tilde) prefix.

**Visit Tags.** After the visit date additional information may be displayed:

1. **days overdue.** If the visit is overdue, and it is possible to compute the number of days overdue, the tag `DOD=#` shows the number of days the visit is overdue (i.e. current report date minus the scheduled due date for the visit). If the visit is overdue but the dates needed to calculate the visit due date are unavailable, the tag `overdue` is shown. The above tags appear by default, but can be altered using the `-tag` option to only tag those visits known to be overdue by a specified number of days, or more.

2. **conditions met.** The following tags identify visits affected by conditions specified in the conditional visit and termination maps:

`CV#[rox]` = conditional visit map entry # and action, where r=required, o=optional, and x=excluded  
`CT#` = conditional termination map entry #

3. **special plates received.** The following tags identify special plates which were received and may affect subsequent scheduling:

`ETP#` = early termination plate # was received at this visit  
`MVP#` = missed visit plate # was received at this visit

4. **study day.** It is possible to display the day on which visits were performed or are scheduled to be performed using the `-tag` option with either `cday` or `sday` (only one is allowed). When these options are used, the day appears as the first tag to the right of the visit date. With the `cday` option study day is computed for each subject from each cycle baseline visit. With the `sday` option study day is computed from the baseline of the first visit received.

5. **off-schedule.** Using the `-tag` option it is possible to tag visits that were performed off-schedule by a specified number of days, or more. The tags have the following format:

`S-#` = visit was performed early by # days  
`S+#` = visit was performed late by # days

## Options

The options for this report are grouped into 4 categories. Subject and visit selection options determine which subjects and visits are included in the report, visit tag options determine which information tags are displayed, and the formatting options allow limited control over the report output format. By default all subjects enrolled in the trial and all visits defined in the visit map are included in the report. If more than one subject and/or visit selection option is specified, only those subjects and visits that meet all of the selection criteria are included in the report.

### Subject Selection Options

- itime e# l# o# oU done| notdone** select the subjects to include based upon one or more of the criteria:
- e#: select subjects with 1+ visits early by # days or more
  - l#: select subjects with 1+ visits late by # days or more
  - o#: select subjects with 1+ visits overdue by # days or more
  - oU: select subjects with 1+ visits overdue, duration unknown
  - done: select subjects with no remaining scheduled or overdue visits
  - notdone: select subjects with one or more required but not yet received visits
- i #, #-#** select subjects by subject ID
- c #, #-#** select subjects by site ID

### Visit Selection Options

- v #, #-#** select by visit number
- vtype list** select by visit type where *list* may contain any combination of the visit type characters from the list X, P, B, O, S, T, W, F, E, A, R, and r.
- vtime e# l# o# oU** select visits performed early or late by # days or more, or overdue by # days or more, or overdue by an unknown duration
- e#: select visits early by # days or more
  - l#: select visits late by # days or more
  - o#: select visits overdue by # days or more
  - oU: select visits overdue, duration unknown
- cneed list** select cycles by their need where *list* may contain any combination of the keywords: unknown, required, next, optional, excluded.<sup>1</sup>
- Each cycle is classified as belonging to only one of these categories. Thus option *required* does not include *next*. To list all required cycles, including the next scheduled cycle for each subject, you must specify both, i.e. `-cneed required next`.
- cdone list** select cycles by their status where *list* may contain any combination of the keywords: notdone, overdue, inprogress, terminate, abort.
- Each cycle is classified as belonging to only one of these categories. Do not be confused by the common English interpretation of these words. For example, option *notdone* does not include *overdue*, and option *terminate* does not include *abort*. To list all cycles that are not done,

<sup>1</sup> Only the first 3 characters of each keyword are significant. For example, `-cneed required next` and `-cneed req nex` are equivalent and will select the next required cycle, plus all other required cycles, whether they have been completed or not. This applies to the keywords supplied for any of the `-cneed`, `-cdone`, `-vneed`, and `-vdone` options.

including those that are overdue, you must specify both, i.e. `-cdone notdone overdue`. And to list all cycles that have ended in either a cycle termination or an abort you must specify both, i.e. `-cdone terminate abort`.

#### **-vneed *list***

select visits by their need where *list* may contain any combination of the keywords: `unknown`, `required`, `next`, `optional`, `excluded`.

Each visit is classified as belonging to only one of these categories. Thus option `required` does not include `next`. To list all required visits, including the next scheduled visit, you must specify both, i.e. `-vneed required next`.

#### **-vdone *list***

select visits by their status where *list* may contain any combination of the keywords: `notdone`, `overdue`, `missed`, `received`, `terminate`, `abort`.

Each visit is classified as belonging to only one of these categories. Do not be confused by the common English interpretation of these words. For example, option `notdone` does not include `overdue`, and option `received` does not include `terminate` or `abort`. Visits registered as missed might be considered not done or done depending on you whether you mean that the visit never occurred or that it is no longer expected. Thus, to list all visits that are not done, including those that are overdue, you must specify both, i.e. `-vdone notdone overdue`, and if you want to include visits that have been recorded as missed the specification will be `-vdone notdone overdue missed`. Similarly, to list all visits that have arrived in the database you must specify `-vdone received terminate abort`.

## Visit Tag Options

#### **-tag all[[lost.mvp/etp/CT/CV/overdue] e# l# o# sday|cday]**

For each visit that is output include the following information, which is selected by including the listed keyword:

`missed`: all required plates are registered as missed  
`mvp`: visits with a missed visit notification plate  
`etp`: visits with an early termination plate  
`CT`: visits on which a conditional termination occurred  
`CV`: visits affected by a conditional visit map entry  
`overdue`: visits currently overdue  
`all`: all of `lost`, `mvp`, `etp`, `CT`, `CV`, and `overdue` (this is the default)  
`e#`: tag visits early by # days or more  
`l#`: tag visits late by # days or more  
`o#`: tag visits overdue by # days or more  
`sday`: show days since arrival of first valid Visit Date  
`cday`: show days since cycle baseline date

## Output Format Options

#### **-pg rows columns 0|1**

set page length, width and one subject/page option. The default is 57 lines by 80 columns and 0 = one page/subject option is turned off.

#### **-fmt fmt**

set output format. The `-fmt` option can be used to change the output format used for each visit. The format statement follows awk/C syntax. The default output format for each visit is:

```
"%10d %5d① %s② %-15.15s %4s %2s %2s %-11.11s③ %s"
```

- ① Print number right justified in 5 spaces (but more if required).
- ② Print string right justified, in as many spaces as required.
- ③ Print string, left justified in exactly 11 spaces (truncate if >11).

Note that there is one specification for each of the 9 output columns: Subject ID, visit#, visit type, visit label, due day, overdue allowance, visit need&status, visit date, and the visit tags. Also note that the entire format string must be enclosed in single or double quotes.

## Examples

**Example 2.37. Select subjects from sites 40-44 for whom any visit was performed early or late by 10+ days, or with a visit currently overdue by 30+ days, or with a visit overdue by an unknown duration**

```
-c 40-44 -itime e10 l10 o30 oU
```

**Example 2.38. Show visits 101-199 for subjects who have completed all follow-up**

```
-itime done -v 101-199
```

**Example 2.39. Show baseline, scheduled follow-up and termination visits only**

```
-vtype BST
```

**Example 2.40. Show cycles that are not expected but which nonetheless are in progress or have been completed**

```
-cneed excl -cdone inpr term abort
```

**Example 2.41. Show all info tags including study day**

```
-tag all sday
```

## Limitations

1. The input for **DF\_PTvisits** comes from summary file `DFX_schedule` (created by **DF\_QCupdate**). Thus, the output will reflect the status of subject follow-up as of the most recent execution of **DF\_QCupdate**, and will not contain any changes that have occurred since then.
2. Visit requirements and scheduling are not fixed and may change during the trial. The requirement for each cycle and visit, and the scheduled due dates may change for each subject due to: new database conditions triggered at follow-up visits, corrections to the data values that trigger conditions, corrections to the dates recorded for baseline or termination dates, or to visits used to schedule subsequent visits.

## See Also

[DF\\_QCupdate](#)  
[DF\\_PTschedule](#)

## Name

**DF\_QCfax** — Fax or email Query Reports

## Synopsis

```
DF_QCfax {study} [-t] [-f #] [-F replyto] [ [-p password] | [-P] ] [-r #] [-d #] [-q #] [-A4] [-font FontName] [-label SignatureLineLabel] [-h hostname] [-w hh:mm] [ [-u report(s)] | [-s report(s)] | [-i name] | [-b yyymmdd] | [-a ] ]
```

## Description

**DF\_QCfax** converts Query Reports into PDF documents, includes DFdiscover bar coding, and transmits them to the clinical sites and/or other destinations by fax or email.

Often, **DF\_QCfax** will be run shortly after creating a new batch of Query Reports, to send them to the clinical sites. When this is the case, it is not necessary to specify the reports to be sent. The default is to send all reports listed in file `reports/QC/QC_NEW`, which is re-created each time **DF\_QCreports** is used to create a new batch of external Query Reports. If some of the reports listed in this file have already been transmitted by a previous run of **DF\_QCfax** they will be skipped, and not sent out again.

When not sending a batch of newly created external Query Reports, the reports to be transmitted must be specified explicitly using one of 5 options, only one of which can be used at a time. The `-s` and `-u` options are used to identified previously sent and unsent external Query Reports respectively, and the `-i` option is used to identify internal Query Reports, whether previously sent or not. Each of these 3 options must be followed by a space delimited list of the report names to be transmitted.

The 4<sup>th</sup> option, `-b`, is used to re-transmit the backlog of transmission failures (if any) from a previous run of **DF\_QCfax** for a set of external Query Reports created on a particular date. The `-b` option is followed by a single report date in `yyymmdd` format. This identifies the creation date of the set of Query Reports to be re-tried if they have not yet been successfully transmitted. (Note: This is the date included in the Query Report name. It is the report creation date and may be different from the date on which the Query Reports were transmitted by **DF\_QCfax**). This is the option to use to ensure that Query Reports are re-transmitted to all of the destinations listed for each clinical site in the sites database. While re-running **DF\_QCfax** with no options will keep trying to send any Query Reports listed in file `reports/QC/QC_NEW` which have not been delivered, it only does this for the primary destination for each external Query Report. The `-b` option will do it for the secondary destinations as well.

The 5<sup>th</sup> and final option, `-a`, is used to transmit all unsent reports, residing in the `QC` directory. Use this option carefully. Each time **DF\_QCreports** is run, Query Reports are created in the study `QC` directory. If any of these reports are not transmitted using **DF\_QCfax**, or marked as sent using **DF\_QCsent**, they remain in this directory. Subsequently running **DF\_QCfax** with the `-a` option will fax *all* of the Query Reports found in the `QC` directory. This could result in sending old Query Reports that are no longer relevant. In many cases the `-b` option will be a better choice.

### Use of `-b`

The backlog option relies on finding a record of all previous transmission attempts in file `dfdiscover.log`, a journal of Query Report transmissions maintained by **DF\_QCfax**. Thus changes to the sites database to modify or add destinations for any of the clinical sites, will not be used by the backlog option. Only destinations registered in `dfdiscover.log` are considered for re-transmission. Changes to the destinations listed in the sites database only become effective for Query Reports sent by **DF\_QCfax** after the changes were made.

**DF\_QCfax** needs to know whether any given transmission is bound for a primary or secondary destination. Thus when the backlog option is used it looks for at least one entry in `dfdiscover.log` with the `PRIMARY` keyword, and will not attempt to re-send any of the `FAILED` Query Reports unless this keyword is found.



When sending external Query Reports to the clinical sites, it is not necessary to specify the destination fax number or email address for each site. By default, **DF\_QCfax** will send the external Query Report created for each clinical site to all of the fax or email destinations listed in the `Primary Fax` field of the sites database for that site. This field can contain one or more fax numbers and/or email addresses. The first fax number or email address in the field is considered the primary destination for the clinical site and all others are secondary destinations. However, it is also possible to override the destination(s) specified in the sites database by using the `-f` option followed by a space delimited list of fax numbers and/or email addresses. With this option, each of the specified reports is sent to all of the specified destinations. Since there is no site ID automatically associated with internal Query Reports, the `-f` option must be used to specify at least one destination fax number and/or email address, when sending internal reports.

When sending an external Query Report to multiple destinations, the first fax number or email address in the list of destinations is considered the *primary destination* for that site, and all others are considered secondary destinations. This is true whether the destination list is specified explicitly using the `-f` option, or is obtained from the `Primary Fax` field of the sites database. On successful transmission of an external Query Report to a primary destination **DF\_QCfax** runs **DF\_QCsent** which marks all of the queries in the report `sent`, and moves the report to the `QC/sent` directory. This does not happen on successful transmission to secondary destinations. Reports which could not be transmitted to their primary destination remain in the `QC` directory, and their queries retain their old status of `in report`.

If a transmission is successful the report name, date, time and fax number is registered in with the keyword `FAXED`, and if it failed it is registered with the keyword `FAILED`. The keyword `PRIMARY` identifies those transmissions bound for the primary destination for a clinical site.

## Emailing Query Reports

Due to the limitations of email technology, it is not always possible to accurately determine the final status of emailed reports, and thus they are always considered to have been successful, and are registered in with the keyword `FAXED`. Since emailed Query Reports are always considered successful, **DF\_QCsent** will always execute if the primary destination has an email address, and thus it is possible that query status may be changed to `sent` and the Query Report moved to the `sent` directory even though the Query Report was not delivered.

It is recommended that a fax number be used as the primary destination for a site whenever one is available, and email address(s) (and additional fax numbers) be used as secondary destinations. Use an email address as the primary destination when it is the only method available.

Since both **DF\_QCfaxlog** and the `-b` option rely on the existence of information in `.`, it is critical that this file not be deleted or modified in any way. Using **DF\_QCfaxlog** to check the transmission status of Query Reports is the safe way to review the records in `.`

If transmission is by email, DFdiscover converts the Query Report to PDF format (if it is not already in that format) and then sends it as a MIME attachment to the specified email address. This PDF attachment can optionally be password protected and encrypted using the `-p password` to specify a password, or the `-P` option to use the user's default password.

When emailing an external Query Report the email return address is taken from the `REPLY TO` field of the sites database. The `-F` option can be used to override this value or to add a return address when emailing an internal Query Report. If a return address is not supplied in either of these ways the email will be identified as coming from user `datafax` and any reply sent to this address will be delivered to the DFdiscover problem mail recipient.

In Query Reports sent via email, the subject line of the email has the following format: `Query Report for studyname: site site #, year/month/day` where `studyname` is determined from `/opt/dfdiscover/lib/DFstudies.db`. If `studyname` cannot be determined, the subject line of the email has the following format: `Query Report for study #: site site #, year/month/day`

By default reports are sent immediately. The `-w` option can be used to delay transmission to a specified time. This option accepts the following values:

- *hh:mm*: the next occurrence of a specified time (using a 24 hr clock)
- *now*: transmit immediately (default)
- *midnight*: transmit at midnight tonight
- *noon*: transmit at noon

The complete list of values which are legal for this option matches those available for **DFsendfax** and can be found in [Programmer Guide, Delayed Sending](#).

## Sending documents manually

If for some reason you decide not to distribute Query Reports using **DF\_QCfax**, but instead to send them to the clinical sites some other way, remember to run **DF\_QCsent** to register these documents as having been sent. Otherwise and the queries will not be updated. As a result, **DF\_QCfaxlog** could not be used to track report transmissions, the **-b** backlog option could not be used, the **-o** of **DF\_QCreports** would not operate as intended, and the true status of queries would not be accurately reflected in **DFexplore** or **DFexport.rpc**.

## Options

- f #** Use the specified destination fax number or email address for all reports. Fax numbers must be specified as you would dial them from your fax machine (including long distance and/or area codes where necessary). Email addresses must be specified in the format **mailto:email\_address**, where **mailto:** is fixed and required, and **email\_address** is a valid email address.
- F replyto** Use the specified return email address for all reports. The email return address must be a fully qualified, valid email address. Do not include the **mailto:** prefix used with the **-f** option. No prefix is required or allowed.
- 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 document to send. The total number of attempts is this number of retries plus 1. The default number of retries is 2.
- d #** Specify the number of minutes to wait before retrying a failed attempt. The default delay wait is 10 minutes.
- q #** Specify the outgoing document queue number to be used for sending. The default is to add all documents to the same queue.
- A4** Format the output for A4-sized paper. The default is to format for US letter-sized paper.
- font FontName** Use the specified font, or the default "Courier New" if the specified font is not found on the DFdiscover server.
- label SignatureLineLabel** Send Query Reports with a customized label for the investigator signature line at the top of each page. Label text must be enclosed in double quotes. Empty double quotes allow the signature line to be omitted from all pages of the report. The default signature line label is "Study Coordinator Sign and Date".
- h hostname** Specify the outgoing HylaFAX fax server hostname. The default is the HylaFAX fax server on the **DFdiscover** master machine.

<b>-w hh:mm</b>	Delay transmission until the specified time (hour and minute)
<b>-a</b>	Fax <i>all</i> unsent Query Reports found in the study <code>QC</code> directory. See the warning before using this option.
<b>-u report(s)</b>	Fax the specified unsent report(s) (from the <code>QC</code> directory)
<b>-s report(s)</b>	Fax the specified previously sent report(s) (from the <code>QC/sent</code> directory). This option is used to resend one or more external Query Reports, which have already been registered as successfully sent to the clinical site(s).
<b>-i name</b>	Fax the named internal report(s) (from the <code>QC/internal</code> directory). This option requires the name(s) of one or more internal Query Reports (from the study <code>QC/internal</code> directory) and a destination fax number specified with <code>-f</code> .
<b>-b yymmdd</b>	Re-send a Query Report to all destinations to which it has so far failed on all attempts. <code>yymmdd</code> represents the date on which the report was created (the second part of the Query Report name). Only one date may be specified on each run of <b>DF_QCfax</b> with the <code>-b</code> option.
<b>-t</b>	Test mode. Echo what would be done without actually doing it. Use this argument to check that everything is set up properly, and that the desired documents have been correctly identified for transmission. The execution commands from <b>DF_QCfax</b> are displayed as the output, but they are not executed. Each Query Report is identified as to where it would have been sent but not when.

## Examples

### Example 2.42. Send two unsent Query Reports, created for sites 120 and 122 on Aug 15/04

```
-u 120-040815 122-040815
```

### Example 2.43. Fax an internal report named test to 555-1212 from dan at 2am

```
-w 02:00 -f 555-1212 -i test -F dan@hotmail.com
```

### Example 2.44. Send an unsent Query Report created for site 123 on June 3/04, retrying once, if necessary, after a delay of 30 minutes

```
-r 1 -d 30 -u 123-040603
```

### Example 2.45. Email a previously sent external Query Report created for site 150 on June 25/04 to dan@hotmail.com.

```
-s 150-040625 -f mailto:dan@hotmail.com
```

### Example 2.46. Re-send all failed Query Reports created on October 3, 2017.

```
-b 041003
```

### Example 2.47. Email the unsent external Query Report created for site 123 on August 8, 2010 to dan@hotmail.com with a customized signature line

```
-u 123-100808 -f mailto:dan@hotmail.com -label "Please review and sign"
```

## See Also

[DF\\_QCfaxlog](#)  
[DF\\_QCsent](#)  
[DF\\_QCreports](#)

## Name

**DF\_QCfaxlog** — Examine the transmission date/time and status of faxed/emailed Query Reports

## Synopsis

```
DF_QCfaxlog {study} [-l #] [-m string] [-s]
```

## Description

**DF\_QCfaxlog** displays journal entries from the study

A journal entry is written to for each Query Report transmission, by fax or email, that is attempted by **DF\_QCfax**. If transmission is attempted to multiple destinations a separate record is written for each destination.

When a Query Report is sent to multiple destinations, the first destination in the list is considered to be the `PRIMARY` destination, and all others (if any) are considered to be secondary destinations. This is true whether the destination list is set in **DF\_QCfax** using the `-f` option or by using the destinations listed in the `Primary Fax` field of the sites database. The keyword `PRIMARY` in a entry identifies a record for a primary destination.

For each destination, whether primary or secondary, a record is written to after all retries have been completed, that shows whether the transmission was successful. The keyword `FAXED` identifies a successful transmission, and the keyword `FAILED` identifies a failure. If the transmission to the primary destination is successful the log also shows the number of queries included in the report, and indicates that they have been updated to status `sent` in the Query database.

### Warning

Query Reports sent by email are always considered successful, because there is no accurate way to determine the ultimate success or failure of an email transmission.

The following excerpt from displays entries for a single Query Report that was successfully faxed to two different fax numbers and one email address.

```
001-170820  FAXED  to PRIMARY    905-555-9999  Mon Aug 20 11:14:38 2017 (marked 8 queries)
001-170820  FAILED to    905-444-1111  Mon Aug 20 11:15:38 2017
001-170820  FAXED  to mailto:dan@hotmail.com Mon Aug 20 11:16:38 2017
```

These entries indicates that the Query Report, named 001-170820, was successfully sent to the first and third destinations, but failed to the second destination. Successful faxing to the first (`PRIMARY`) destination also caused the 8 queries in this report to be marked as sent.

If a Query Report is successfully sent to the primary destination, but it contains no queries, the log entry would indicate (marked 0 queries).

Records can be selected from with the `-l` and `-m` options. If both selection criteria are used, they are applied in the order specified; altering the order can produce different results, as illustrated by the last 2 examples below. If no selection criteria are specified, all reports listed in `QC_NEW` (i.e. reports created by the last execution of **DF\_QCreports**) are selected.

## Options

**-l #**            Include only the last # lines

**-m string**      Include only lines which match on the specified string; matching is case sensitive

`-s` Sort the output by report ID rather than chronologically (default)

## Examples

### Example 2.48. Display the last 5 lines from

```
-l 5
```

```
DF_QCfaxlog: Transmittal Status for Query Reports.      DFstudy 7. Apr 01,2017 14:55
013-170322  FAXED  to PRIMARY  ###-####  Wed Mar 22 13:34:43 2017  (marked 100 queries)
013-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 13:06:51 2017  (marked  6 queries)
026-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 13:45:24 2017  (marked  4 queries)
009-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 14:16:56 2017  (marked 37 queries)
010-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 15:27:42 2017  (marked 84 queries)
```

### Example 2.49. Display the status for all Query Reports created on 17/03/23, sorted by report ID

```
-s -m 170323
```

```
DF_QCfaxlog: Transmittal Status for Query Reports.      DFstudy 7. Apr 01,2017 15:00
009-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 14:16:56 2017  (marked 37 queries)
010-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 15:27:42 2017  (marked 84 queries)
013-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 13:06:51 2017  (marked  6 queries)
026-170323  FAXED  to PRIMARY  ###-####  Thu Mar 23 13:45:24 2017  (marked  4 queries)
```

### Example 2.50. Display the last 20 FAILED transmissions

```
-m FAIL -l 20
```

### Example 2.51. Display any FAILED transmissions among the last 20 transmissions registered

```
-l 20 -m FAIL
```

## See Also

[DF\\_QCfax](#)

## Name

**DF\_QCprint** — Print Query Reports

## Synopsis

```
DF_QCprint {study} { [-u report(s)] | [-s report(s)] | [-i filename] } [-A4] [-font FontName] [-label SignatureLineLabel]
```

## Description

**DF\_QCprint** formats the specified Query Reports into a PDF document including DFdiscover barcoding and sends the reports to the study printer.

Reports to be printed may reside in one of three directories. The three directories, all under the study reports directory are:

- `QC`: holds Query Reports marked as unsent (or not yet sent)
- `QC/sent`: holds Query Reports previously marked as sent (using **DF\_QCsent**)
- `QC/internal`: holds internal Query Reports

If no options are specified, the reports listed in `QC_NEW` (that is, those reports created by the last execution of **DF\_QCreports**) are printed.

## Options

<b>-u report(s);-s report(s);-i filename</b>	Selects for printing, respectively, the specified <ul style="list-style-type: none"> <li>• unsent report(s) (from the <code>QC</code> directory)</li> <li>• sent report(s) (from the <code>QC/sent</code> directory)</li> <li>• internal report from the <code>QC/internal</code> directory</li> </ul>
<b>-A4</b>	Format the page for A4-sized paper instead of the default US-letter size
<b>-font FontName</b>	Use the specified font, or the default "Courier New" if the specified font is not found on the DFdiscover server.
<b>-label SignatureLineLabel</b>	Print Query Reports with a customized label for the investigator signature line at the top of each page. Label text must be enclosed in double quotes. Empty double quotes allow the signature line to be omitted from all pages of the report. The default signature line label is "Study Coordinator Sign and Date".

## Examples

### Example 2.52. Print specified unsent Query Reports

```
-u 120-910815 122-910815
```

### Example 2.53. Print all sent Query Reports created on 99/04/15

```
-s *-990415
```

**Example 2.54.** Print the unsent external Query Report created for Site 123 on August 8, 2010 with a customized signature line label.

```
-u 123-100808 -label "Please review and sign"
```

## See Also

[DF\\_QCfax](#)



## Name

**DF\_QCreports** — Create Query Reports

## Synopsis

```
DF_QCreports {study} [-i srqQPRc] [-b v/p] [-c #, #-#] [-C #, #-#] [[-cd yy/mm/dd-yy/mm/dd] | [-cd ] ] [-f name] [-q types,tag] [-o #] [-p 0/1/2/3] [-e b/f/b1/f1/x] [-l lastv/lastsv] [-s no/yes] [-P #, #-#] [-S #, #-#] [-T #, #-#] [-v #, #-#] [-V #, #-#] [-F visitlist:platelist:fieldlist] [-N "string1" "string2" ... ] [-idfmt #] [-dtfmt fmt] [-probfmt 1/2] [-rline] [-rspace]
```

## Description

**DF\_QCreports** is used to create reports containing subject scheduling information and data queries. It can be used to create both External Query Reports, designed for transmission to the clinical sites, and Internal Query Reports, which might contain subject data from more than one clinical site or include internal data queries not intended for the clinical sites.

DFdiscover Query Reports can be created with any or all of the following content:

- **Subject Status Summary.** lists subjects and shows entry, last and next assessment dates
- **Fax/Refax List.** lists data queries which require correction and retransmission of CRFs
- **Question & Answer List.** lists data queries which require a written response
- **Cover sheets and messages.** optional information which can be customized for each clinical site

This chapter describes the 2 types of Query Reports, the content and appearance of each section of a Query Report, the study configuration files on which Query Reports depend, the many options available to control the content and format, and some important limitations and recommendations.

## External & Internal Query Reports

External Query Reports are designed for transmission to the clinical sites participating in the trial. They are named using the site ID and creation date in *ccc-yyymmdd* format. For example, 012-170715, 5120-170715 and 21460-170715 identify external Query Reports created for sites 12, 5120 and 21460 on July 15, 2017. This identifier is printed at the top of each page of the report along with the name of the individual responsible for Query Reports at the clinical site and the name of the site, as specified in the sites database. When a Query Report is created an ASCII file containing the report contents is written to the study reports/QC directory. When Query Reports are printed using **DF\_QCprint** or sent to the clinical site using **DF\_QCfax** they are first converted to PDF for a more attractive appearance. Each external Query Report that is successfully transmitted to its clinical site is moved into the study reports/QC/sent directory, where it will remain unless removed manually. If an external report is never transmitted to its clinical site it will remain in the reports/QC directory unless removed manually.

Internal Query Reports are assigned a name by the user when they are created and are written to the reports/QC/internal study directory. If a name is reused the previous version of the report will be overwritten. Like external Query Reports, internal Query Reports may also be printed using **DF\_QCprint** and faxed using **DF\_QCfax** but because the report name does not automatically link them to a specific clinical site, the way external Query Report names do, a destination fax phone number must be specified.

## The Contents of Query Reports

Query Reports have 3 main sections, the subject status list, the refax list and the question and answer list. They may also include a 1 page cover sheet.

## The Subject Status List

This section of the report lists the subjects in the study and provides 3 subject scheduling time points. Each subject is described on a single line starting with the subject ID, which is tagged with an asterisk if the subject has any outstanding data queries, followed by the entry, last follow-up and next follow-up visits. An example is shown below.

### Example 2.55. The Subject Status List

```

QUERY REPORT # 003-040323-01 ( Jane Smith, Dundas Valley Health Clinic )

SUBJECT STATUS SUMMARY (* identifies subjects with data queries in this report)

SUBJECT          ENTRY VISIT          LAST FOLLOW-UP          NEXT FOLLOW-UP
1081*          SCREEN #0: DEC 03,2016      Visit 06: JAN 26,2017      Visit 07: APR 26,2017
1082*          SCREEN #0: not done          Visit 05: MAR 24,2017      Visit 06: APR 23,2017
1083*          SCREEN #0: unknown          Visit 02: unknown          Visit 03: ?
1084*          SCREEN #0: MAR 08,2017      Visit 03: APR 07,2017      Visit 04: MAY 07,2017
1085*          SCREEN #0: overdue          Visit 03: APR 28,2017      Visit 04: MAY 28,2017
1086*          SCREEN #0: APR 15,2017      Visit 04: MAY 15,2017      : done
1087*          SCREEN #0: APR 22,2017      SCREEN #0: APR 22,2017      SCREEN #1: APR 29,2017
1088*          SCREEN #0: APR 26,2017      SCREEN #0: APR 26,2017      SCREEN #1: MAY 03,2017
TOTAL CASES = 8
    
```

Each visit is identified by its page map label (as in the above example) or by the visit number if a label is not specified for the visit in the page map.

The visit displayed as the entry visit can be the first screening visit (as in the above example), or the first scheduled or baseline visit from the first or most recent cycle. This is selected using the `-e` option.

The last visit can be either the last scheduled visit or the last completed visit, whether it was scheduled or not. This is specified using the `-l` option.

The next follow-up visit is the next scheduled visit, whether required or optional. Normally all scheduled visits are required, but it is possible, using the conditional visit map, to indicate that a scheduled visit is optional. The date format corresponds to the format used for visit dates, but this can be overridden using the `-dtfmt` option.

If the entry or last follow-up visit has been completed but the visit date is not available, the phrase `unknown` is used. If the entry or last follow-up visit has been registered as missed, or a missed visit plate has been received for the visit, the phrase `not done` is used. If the entry visit has not yet been received, the phrase `overdue` is used if the visit is overdue, and the phrase `pending` is used if the visit is not yet overdue. If the next follow-up visit date can not be calculated for any reason a `?` is displayed. The phrase `done` is used when there is no next visit because all follow-up has been completed, or follow-up was terminated early.

## Fax/Refax List

This section of the report lists missing CRF pages, overdue visits, and data queries requiring a CRF correction or clarification. All of these problems are resolved by having the clinical sites fax the missing or corrected CRF pages to the DFdiscover system. In the case of corrected CRF pages, DFdiscover recognizes that the page is being resubmitted and allows the data management staff to enter the data corrections and resolve the data queries. All CRF images, old and new, are retained by DFdiscover and are available for review at any time. An example of this section of the Query Report is shown below.

### Example 2.56. The Fax/Refax List

```

FAX/REFAX LIST (Please locate/correct and then fax the following pages of the CRF)
    
```

BE SURE TO INITIAL AND DATE ALL CHANGES.

SUBJECT	CRF FORM & VISIT	PROBLEM
2035	PLT 6, SEQ 0	(Missing Page)
2035	Form 2	Pulse beats/minute = (Missing Value)
2035	Form 5, 3 Month	Other Medication names = (Inconsistent) This field should not be blank given the answer to the previous question.
2035	Form 5, 4 Month	Number of Pills Returned = 45 (Other Problem) This is the number of pills dispensed. If the patient took none of the study medication a compliance problem report is required.
2035	Form 8, Page 1	1. Date of Change = 00/JAN/2017 (Illegible) Please print the day more clearly, was it 17 or 12?
2049	Form 4, Page 1	2. Start Date or Duration = 02/000/2018 (Illegal Value) This is not a valid date. Please correct or clarify.
2049	3 Month Follow-up	(55 days Overdue)
2049	4 Month Follow-up	(25 days Overdue)

The default fax/refax list, shown above, lists all queries for all subjects in a compact format that identifies the subject ID, the CRF page and/or visit and the problem. The CRF page and visit label come from the page map for missing pages and data queries, and from the visit map for overdue visits. If a label is not specified the plate and visit/sequence number are used. The problem type appears in brackets. For missing pages and overdue visits this is all that is needed, but for queries related to data fields the field must also be identified. This is done using the variable description as specified in the study schema defined using the **DFsetup** tool. The field description is followed by the current value of the field from the study database, if there is one. This is followed by the problem type and then by the query or explanation if one exists. In some cases, e.g. a missing value, the problem type (in brackets) is all that is needed to describe the problem and no additional query exists.

Overdue visits and missing pages are identified by **DF\_QCupdate** and thus will only be accurate as of the the most recent execution of **DF\_QCupdate**. For overdue visits the number of days the visit is currently overdue is shown if the visit due date is known. This is calculated using the visit due date calculated by **DF\_QCupdate** and the current date obtained from the computer system clock on the date the report is run.

**DF\_QCreports** includes a number of options that can be used to control the content and format of the fax/refax list. In the example shown above the `-rline` option was used to draw a solid line between subjects. In addition, queries can be selected by site, subject, visit, plate, type and validation level, and each subject's queries can be put on a separate page.

## Question and Answer List

This section of the report lists data queries which ask a question that requires a written response. In some cases these question and answer (Q&A) queries are created because we don't know which CRF page needs to be fixed and refaxed, and we need to know which page was corrected to resolve the problem (see the first query below). In other cases a Q&A query may ask for information that can not be collected by simply correcting and refaxing a CRF page (see the second query below).

### Example 2.57. The Question and Answer List

QUESTION & ANSWER LIST (Print your answer below each question and fax back this page)

SUBJECT	CRF FORM & VISIT	PROBLEM
2035	Form 2 Baseline	Date of Birth = 30/05/48 (Inconsistent) Age (53) on Form 1 does not match age (52) calculated

from date of birth (30/05/48) on Form 2. Please explain the correction below and fax back this report along with the corrected CRF page.

2035	Form 7, 3 Month	Investigator's Signature = J. Smith (Illegal Value) J. Smith is not a registered investigator. Please explain.
------	-----------------	---

## Titles, Cover Sheets and Messages

This section of the report is used to specify a cover sheet containing headings and messages that can be customized for each site. In the example shown below the report title has been customized to include the date and time the report was created as well as the intended report recipient. The cover sheet also includes a general message related to an upcoming meeting, which would be included in the report for all clinical sites, and another message that would only be included in the report sent to those sites for whom this message was currently relevant. This is done using the `QCtitles`, `QCcovers` and `QCmessages` files.

### Example 2.58. Titles, Cover Sheets and Messages

STUDY REPORT # 001-170724-01 for: Jul 24,2017 at 03:55:34 PM  
TO: Jane Smith, Dundas Valley Medical Center

**PLEASE NOTE:**

The next BLOOD PRESSURE Study investigator's meeting will be held on Oct. 9-10, 2017 in Orlando, Florida.  
Location and times will follow at a later date.

**IMPORTANT!**

We have not received any corrections related to the Quality Control reports we have faxed to your site over the last 2 months. Please make every attempt to resolve the outstanding data queries as soon as possible.  
By protocol and agreement with the FDA, all adverse events must be reviewed by us within 3 days of each subject assessment.  
If you are having problems of any kind, please let us know.

## Configuration Files Used By Query Reports

### Subject Scheduling Specifications

The visit map and other scheduling configuration files described in [Study Setup User Guide, Subject Visit Scheduling](#) are used by `DF_QCupdate` to generate the scheduling information and queries for overdue visits and missing CRF pages, reported by `DF_QCreports`.

### The Page Map

By default Query Reports identify queries by plate and sequence numbers. These numbers can be replaced by more meaningful user-defined labels using `DFpage_map`, which can be created using the Page map view in `DFsetup`. The purpose

and use of `DFpage_map` is fully described in [Study Setup User Guide, Page Map](#), and its format is described in [Programmer Guide, DFpage\\_map - page map](#).

## The Sites Database

The sites database specifies the subject IDs belonging to each clinical site and the contact person, fax or email address and institution name for each site, all of which are used in creating Query Reports. It is important that the sites database be accurate and up to date. Always execute the DFdiscover report **DF\_QCupdate** prior to creating Query Reports to ensure that the most recent sites database information is being used. Site information can be entered using the Sites view dialog in **DFsetup**, as described in [Study Setup User Guide, SitesSites](#), and its format is described in [Programmer Guide, DFcenters - sites database](#).

## Customized Query sort order

Queries normally appear in a Query Report sorted by ascending subject ID, by ascending visit number, by ascending plate number, and finally by ascending data field number. This sort order can be over-ridden through `DFqcsort`. If defined, `DFqcsort` is used when Query Reports are created. `DFqcsort` can be created through the Sort map view dialog in **DFsetup**.

The format of `DFqcsort` is described in [Programmer Guide, DFqcsort - Query and CRF sort order](#). The following listing summarizes some of the capabilities:

```
12 | * | 1    ❶
* | 1 | 2    ❷
11 | 2 | 3    ❸
13 | 2 | 4    ❹
5 | 2 | 5     ❺
* | 2 | 6     ❻
* | * | 7     ❼
```

- ❶ Queries on plate 12 of any visit appear first.
- ❷ All queries for visit 1 appear next (order by plate number).
- ❸ Then plate 11 of visit 2.
- ❹ Then plate 13 of visit 2.
- ❺ Then plate 5 of visit 2.
- ❻ Then all remaining plates of visit 2 (ordered by plate number).
- ❼ Finally, any queries that do not match any previous pattern are sorted in the default order of ascending visit number followed by ascending plate number.

\* may be used to match all visits for a specified plate, or to match all plates for a specified visit. In such cases queries are sorted on the unspecified field within the specified field. For example,

```
12 | * | 1
```

will sort all queries on plate 12 first. If such queries appear on plate 12 for 2 or more visits they will appear sorted by visit number.

Control is provided over plate and visit sort order only. Queries are always ordered by ascending subject ID, and by ascending data field number within each plate.

## Query Report Titles

The standard title at the top of each page and above each of the 3 sections of the report can be customized through the use of the file `QCtitles` in the study `lib` directory. This file can be created using the Query Titles view in **DFsetup**. **DF\_QCreports** checks for the existence of this file and will use it if it exists; otherwise, the standard titles are used.

The following example of a valid Qctitles file illustrates some of the capabilities:

```
# TITLE AT THE TOP OF EACH PAGE OF AN EXTERNAL QUERY REPORT
<EXTERNAL font="^P12 fB">
ACE STUDY REPORT <NUM> for: <MON> <DAY>,<YEAR> at <TIME>
TO: <WHO>, <WHERE>
</EXTERNAL>

# TITLE AT THE TOP OF EACH PAGE OF AN INTERNAL QUERY REPORT
<INTERNAL font="^P12 fB">
INTERNAL QUERY REPORT: <NAME> page <PAGE> created: <YEAR>--<MON>--<DAY>
</INTERNAL>

# SUB-TITLE ABOVE SUBJECT STATUS LIST (Part 1 of a standard external Query Report)
<STATUSLIST font="^P10 fB">
SUBJECT VISIT SCHEDULE: Note: * identifies subjects with data queries.
</STATUSLIST>

# SUB-TITLE ABOVE REFAX LIST (Part 2 of a standard external Query Report)
<REFAXLIST font="^P10 fB">
CRF CORRECTIONS: Initial and date each correction. Re-fax all corrected pages.
</REFAXLIST>

# SUB-TITLE ABOVE QUESTION & ANSWER LIST (Part 3 of a standard external Query Report)
<QALIST font="^P10 fB">
QUERIES: Print your answer below each question and fax back this page.
</QALIST>
```

The following rules are relevant to the Qctitles file:

- Title specifications must be formatted exactly as shown.
- The opening and closing tags for each section must appear on new lines by themselves with no leading or trailing text outside of the tag. Nothing can precede the "<". No space is allowed within the opening and closing tag except before the word `font`.
- Anything entered outside of a tagged block is ignored.
- A # character may be used to indicate a comment line but # is not strictly needed, and has no special meaning inside a tagged block. For example, it is legal to include a line of # as part of a title inside a tagged block.
- Tags are only recognized if they begin the new line; nothing can precede the "<". No space is allowed within the opening and closing tag except before the word `font`.
- The font value must be enclosed in double quotes. Limited font specifications may be included within the opening tag. ^P (control P, not circumflex P) sets the point size (^P12 and ^P10 are recommended). Available font types include:

fB for bold

fH for normal helvetica

fCW for constant width

The font specification is optional. Without it, `font="^P10 fCW"` is used for internal and external report titles and `font="^P10 fB"` is used for the sub-titles for the three sections of the report.

The following variables may be included in the external and internal report titles which appear at the top of each page, but not in the sub-titles for the 3 parts of the report (status, refax and question & answer lists). The study name comes from

the study configuration file maintained by **DFadmin**. The variables that describe clinical sites come from the sites database file maintained in the Sites view of **DFsetup**.

**Table 2.2. Variables available for use in QCcovers, QCmessages, and QCtitles**

Variable	Use in External	Use in Internal	Meaning
<STUDYNAME>	yes	yes	study name
<SITE> <sup>a</sup>	yes	no	site ID
<WHO>	yes	no	primary site contact
<WHERE>	yes	no	site affiliation
<MAIL>	yes	no	site mailing address
<PRIMEFAX> <FAX1>	yes	no	primary fax number (to which Query Reports are sent)
<FAX2>	yes	no	secondary fax number
<PHONE1>	yes	no	primary contact's phone number
<PI>	yes	no	principal investigator
<PHONE2>	yes	no	principal investigator's phone number
<REPLYTO>	yes	no	email address to which replies made to emailed Query Reports will be sent
<NUM>	yes	no	external report name composed of site ID, date (yymmdd), and page, e.g. 025-080115-01.
<NAME>	yes	yes	external report name composed of site ID and date only, e.g. 025-080115
<PAGE>	yes	yes	page number of Query Report
<DAY>	yes	yes	two-digit day of month
<MON>	yes	yes	three-character month of year
<YEAR>	yes	yes	four-digit year
<WKDAY>	yes	yes	three-character day of week
<TIME>	yes	yes	time of day (hh:mm:ss AM/PM), e.g. 01:12:22 PM
<DATE>	yes	yes	date (ddd mmm dd hh:mm:ss yyyy), e.g. Tue Jan 15 14:34:07 2018

<sup>a</sup>The variable <CENTER> is also supported for backwards compatibility

## Cover sheets

External report cover sheets can be included by defining them in the file `QCcovers` in the study `lib` directory. This file can be created using the Query Covers view in **DFsetup**.

A cover sheet begins with a

```
<FOR site="#list">
```

tag to identify the site(s) that are to receive the cover sheet whose definition follows. It is legal for some sites to receive cover sheets while others do not, and for different sites to receive different cover sheets.

The body of a cover sheet is defined in a <TEXT> block. Within <TEXT> blocks, variable substitutions may be performed as previously described for customized report titles. Aside from variable substitution, all other text appears exactly as formatted

within the text block. Blank lines used to double space text will be preserved. The default font for cover sheet text is constant width.

### Example 2.59. English and French version of a cover sheet

Sites 20-29 use the French version of the cover sheet, while all other sites use English.

```
<FOR site="1~19,30~300">
<TEXT font="^P12 fB">
```

---

PLEASE DELIVER THIS DOCUMENT TO THE STUDY COORDINATOR

TO: <WHO>

Site <SITE>: <WHERE>

Fax <PRIMEFAX>

<DATE>

</TEXT>

</FOR>

```
<FOR site="20~29">
```

```
<TEXT font="^P12 fB">
```

---

DONNEZ CE RAPPORT AU COORDINATEUR DE RECHERCHE, S'IL VOUS PLAÎT

TO: <WHO>

Centre <SITE>: <WHERE>

Fax <PRIMEFAX>

<DATE>

</TEXT>

</FOR>

If more than one cover sheet is defined and a site ID matches more than one inclusion criteria specified by `<FOR site="#list">`, the TEXT blocks from all matching cover sheet specifications are concatenated to form the Query Report cover sheet for that site.

## Messages

Cover sheets may also include messages for all or specified sites. Messages are stored in `QCmessages` of the study `lib` directory and follow the same rules as `QCcovers`. It can be created using the Query Messages view in **DFsetup**.

While messages could be added to cover sheets by putting them in TEXT blocks within `QCcovers`, the use of a separate file for messages makes it easier to keep the fixed, probably unchanging, cover sheet header separate from the messages which will likely change throughout the trial.

`QCmessages` may include more than one message, and each site may receive none, some, or all of them, as selected by the

```
<FOR site="#list">
```



tag which must come at the beginning of each message. If a site is to receive more than one message, the messages will be printed on the cover sheet in file order.

It is not necessary to define a cover sheet in order to use messages. If a message is specified for a site which does not have a cover sheet, a blank cover sheet will be created onto which the message(s) will be printed. The default font for message text is constant width.

### Example 2.60. Typical use of QCMessages

All sites (1~300) receive the announcement of an upcoming meeting. For sites 1, 4, 22 to 24, and 127 this is followed by an important message regarding their lack of response to data queries.

```
<FOR site="1~300">
<TEXT font="^P12 fB">
```

```
PLEASE NOTE:
```

```
</TEXT>
```

```
<TEXT font="^P10 fCW">
```

```
The next study investigators meeting will be held at the time of the
American Heart Association meeting in Orlando, Feb 12-15, 2018.
```

```
The exact date and location is yet to be set.
```

```
</TEXT>
```

```
</FOR>
```

```
<FOR site="1,4,22~24,127">
```

```
<TEXT font="^P12 fB">
```

```
IMPORTANT!
```

```
</TEXT>
```

```
<TEXT font="^P10 fCW">
```

```
We have not received any corrections related to the Query Reports we have faxed
to your site over the last 2 months. Please try to resolve the outstanding
data queries as soon as possible. Some of this information is very important.
```

```
By protocol and agreement with the FDA, all adverse events must be reviewed
by us within 3 days of each subject assessment.
```

```
If you are having problems of any kind please let us know.
```

```
</TEXT>
```

```
</FOR>
```

## Options

**-i srqQPRc**

Select the parts to be included in each Query Report.

s: Subject 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  
 R: Refax notes only appear 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), Subject Status Summary, Fax/Refax List, and Question & Answer List.

The `P`, `Q`, and `R` 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 subject. The `Q` flag creates a new page for each subject for Q&A notes only. The `R` flag creates a new page for each subject for Refax notes only. Thus for example `-i srQ` will create the standard Subject Status Summary and Fax/Refax List followed by a separate page(s) for each subject who has unresolved Q&A notes, and `-i sP`, will produce reports containing the subject status list for all subjects followed by a separate page(s) of queries for each subject who has outstanding unresolved queries. Some combinations, e.g. `-i rqP`, represent an inconsistent request and are thus illegal.

Cover sheets and messages can be generated without a Query 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 PQR` by placing a page break between visits within subjects (`-b v`), or by placing a page break between plates within visits within subjects (`-b p`).

**-c #, #-#**

Include only the specified site IDs.

**-C #, #-#**

Include only the specified subject IDs. Only create reports for study sites that include the specified subjects, and include only the specified subjects in those reports.

**-cd yy/mm/dd-yy/mm/dd-cd**

Select only queries created within the specified date range, inclusively.

**-f name**

Create a single internal Query Report (not for transmission to study sites) and save it to the named file in the `$REPORTS_DIR/QC/internal` directory. All selected queries are included in the same internal Query Report. If a query 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 queries by type. Any combination of the following options may be specified:

I: all internal queries  
 E: all external queries  
 R: all resolved queries  
 U: all unresolved queries  
 P: all pending queries  
 iu: internal unresolved queries  
 ir: internal resolved queries  
 ip: internal pending queries  
 eu: external unresolved queries  
 er: external resolved queries  
 ep: external pending queries  
 all: all internal and external queries 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 queries).
- The keyword `tag` may also be used with the `-q` option to show the status and usage of each query in the report. If `tag` is specified, each query 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 query will be tagged as `(Missing Value-eu)` in the internal Query Report.
- o #** Include old queries only if they previously appeared in an external Query Report that was created more than the specified number of days ago. The default is 0 which includes all unresolved queries in every Query Report.
- p 0|1|2|3** Limit the Subject Status Summary to specified subjects from the following types:
- 0, all subjects (the default)
  - 1, subjects with ongoing follow-up
  - 2, subjects with unresolved queries
  - 3, subjects with ongoing follow-up, unresolved queries, or both
- e b|f|b1|f1|x** Set the "Entry Visit" displayed on the Subject 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 Subject 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 subjects on the Subject Status Summary who only have screening visits (no or yes)?
- The default for this option is `yes`, meaning that all subjects will be included. If `no` is selected those subjects 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 queries on the specified plates
- S #, #-#** Include only queries on specified visits
- T #, #-#** Include only queries 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, 30-99=user-defined problem type
- v #, #-#** Include only queries on data records that are at the specified validation levels. This does not apply to missing plate or overdue visit queries. These queries are always output.
- V #, #-#** Include only queries which are themselves at the specified validation levels. This does not apply to missing plate or overdue visit queries. These notes are output regardless of their validation level.
- F visitlist:platelist:fieldlist** Select queries 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 queries 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 queries are included in the Query 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 subject IDs to the requested number of digits. The default is no zero padding of subject IDs.
- dtfmt fmt** Set the output format for dates displayed in the Subject Status Summary.
- The date format can be set as illustrated in the following examples for Jan 15, 2018:
- `-dtfmt "mm/dd/yy"` displays 01/15/18
  - `-dtfmt "dd/mm/yyyy"` displays 15/01/2018
  - `-dtfmt "yyyy/mmdd"` displays 2018/0115
  - `-dtfmt "mmm dd, yyyy"` displays Jan 15,2018
  - `-dtfmt "yyyymmmdd"` displays 2018Jan15
- 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.
- probfmt 1|2** Number queries within each Query Report and put the problem description first, rather than at the end of the line, where
- 1 the default format is preserved (queries are not numbered and the problem description follows the subject ID, plate and sequence number).
  - 2 queries are numbered and problem description appears first in the description of each query.
- If `-i P` is used to put all queries for each subject on the same page(s), `-probfmt 2` will number the queries within subject ID, otherwise all queries will be numbered in a single sequence within each Query Report.
- rline** Draw a line between subjects in the refax list; otherwise, subjects are separated by a blank line.
- rspace** 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.

## Examples

**Example 2.61. Create reports with only Fax/Refax List and Question & Answer List sections for sites 15 and 21-24**

```
-c 15 21~24 -i rq
```

**Example 2.62. Create reports that include unresolved queries sent >30 days ago**

```
-o 30
```

**Example 2.63. Create an internal subject status report**

```
-f stat.911015 -i s
```

**Example 2.64. Exclude completed subjects with no outstanding queries**

```
-p 3
```

**Example 2.65. Create an internal subject status report using the first visit from the most recent cycle**

```
-f temp -i s -e f
```

**Example 2.66. Create an internal report including all internal queries plus external unresolved queries**

```
-f sum -q I,eu
```

**Example 2.67. Include the Fax/Refax List and Question & Answer List sections (but not the status list) and put Q&As on a new page for each subject**

```
-i rQ
```

**Example 2.68. Include a cover sheet and the Subject Status Summary for all subjects, and put all Refax queries only on a new page for each subject**

```
-i csR
```

**Example 2.69. Include the Subject Status Summary for all subjects followed by a new page for each subject with unresolved queries, with both the refax and Q&A notes together on a separate page for each subject**

```
-i sP
```

**Example 2.70. Include the Fax/Refax List and Question & Answer List queries (but not the Subject Status Summary) together on a separate page for each subject. Sequentially number the queries within subject ID and place the problem description first for each query.**

```
-i P -probfmt 2
```

**Example 2.71. Create a report for sites 10 through 15 only, containing only the cover sheet**

```
-c 10~15 -i c
```

**Example 2.72. Include only queries which are on plates 1 and 2 with sequence number 0 and which have reached validation levels 4 through 7**

```
-v 4~7 -S 0 -P 1,2
```

**Example 2.73. Create reports listing only the Fax/Refax List and Question & Answer List sections and that include only problem types 3, 4 and 21**

```
-T 3-4,21 -i rq
```

**Example 2.74. Create reports listing only refax requests on plates 1 through 13 inclusive that were created between April 1 and April 28, 2018**

```
-cd 18/04/01~18/04/28 -i r -P 1~13
```

**Example 2.75. Create reports for sites 11 and 23 through 30 inclusive that include only the Fax/Refax List and Question & Answer List sections**

```
-c 11 23-30 -i rq
```

**Example 2.76. Create an internal report named `int` that contains query categories 1 through 6, refax and Q&A queries, for visit 3 and plate 11 only**

```
-f int -T 1-6 -i rq -S 3 -P 11
```

**Example 2.77. Create an internal report named `int` that contains internal, unresolved queries of refax type**

```
-f int -q iu -i r
```

**Example 2.78. Create a report for site 10 that includes only the Subject Status Summary section, subjects with ongoing follow-up, the entry visit as the baseline visit, and the last follow-up visit as any visit that has a `Visit Date`**

```
-c 10 -e b -l lastv -p 1 -i s
```

To create the same report but using the last scheduled visit as the last follow-up date, use:

```
-c 10 -e f -l lastsv -p 1 -i s
```

**Example 2.79. Create a standard 3-part Query Report for all subjects of Site 5 that have external, unresolved queries on field 9 of visit 0, plate 1, on all fields of visits 1-20, plate 5, and on all fields of all plates at visit 50.**

```
-c 5 -F 0:1:9 + 1-20:5: + 50::
```

**Example 2.80. Create an internal Query Report named `monitor_rpt` that includes only internal queries having either or both of the strings `PI` or `COORD` in their `Note` field.**

```
-q I -N "PI" "COORD" -f monitor_rpt
```

## Limitations and Recommendations

The following points are important to understand.

- **The content of Query Reports depend on `DF_QCupdate`.** `DF_QCreports` does not calculate the dates needed for subject scheduling or identify the plates and visits that are currently missing and overdue. All it does is report the results created by `DF_QCupdate` and stored in file `DFX_schedule` in the study `work` directory. The only calculation `DF_QCreports` performs is to update the number of days overdue for overdue visits recorded in `DFX_schedule`. Thus the reports created by `DF_QCreports` will only be as accurate as `DFX_schedule`. It is recommended that `DF_QCupdate` be run immediately before running `DF_QCreports` when creating external Query Reports for transmission to the clinical sites.
- **Before running `DF_QCupdate` check internal consistency.** `DF_QCupdate` has dependencies of its own and will only produce accurate results if:
  - the visit map and other configuration files used by `DF_QCreports` are up to date
  - all data received to date has been validated to at least level 1 in the study database
  - the key fields (ID, Visit, Plate) are correct for all records in the database
  - the `VisitDate` fields do not contain errors.

Corrections made to any of the files on which `DF_QCreports` depends take effect with the next batch of reports created after the corrections are made. However, since `DF_QCreports` only reads the output from `DF_QCupdate` it will generally be necessary to re-run `DF_QCupdate` as well. Before running either of these programs corrections should be checked for possible errors using `DF_ICvisitmap` and `DF_ICschema`.

If data remains in the new queue the Query Reports will not be accurate because some subject assessments, CRF plates and replies to previous data queries will not have been registered in the study database. As a result visits may be called overdue and plates may be called missing when in fact the sites have already submitted them. And queries, to which the sites have already responded, may be repeated. This can produce confusion and annoyance and should be avoided by scheduling the creation and transmission of external Query Reports at a time when all data received will have been entered.

If the database contains errors in the key fields, a visit or plate that belongs to one subject might be attributed to another, or to a subject that doesn't even exist, and incorrectly recorded visit dates can result in subject scheduling errors. Such errors are the bane of data management and procedures must be adopted to minimize their occurrence and catch and correct them when they occur. It is recommended that `DF_ICkeys`, `DF_ICvisitdates` and `DF_PTunexpected` be used before running `DF_QCupdate` to help identify such problems.

- **Database restrictions may limit a user's ability to create Query Reports .** If a user has restricted database access, their ability to create Query Reports may be limited. The user is warned of such restrictions, if they exist, during execution of `DF_QCreports`.
- **Create and send at most one external Query Report per site each day.** External Query Reports are named by site ID and date. As a result, there can be only one external Query Report per site per day. If `DF_QCreports` is run more than once on the same date it will overwrite any existing external Query Reports with the same name. It is not a problem



to create an external Query Report, review it, perhaps correct some of the data queries, and then recreate the Query Report before sending it to the clinical site. However, before recreating external Query Reports on the same day there are 2 important limitations you must keep in mind.

- If you use any of the selection options to limit which queries are written to the reports make sure you use the same options each time you recreate a report, or if you change the options, make sure you do so in a way that does not narrow the query selection criteria. Here's why.

When an external Query Report is created, the query fields which identify the current external report for each query are updated. These updates are never rolled back! Thus for example, if a report is created that contains all unresolved external queries, and then is recreated the same day using options that limit which queries are written to the new version of the report, some queries already registered as being in that report may not appear in the final version of the report created that day. To make matters worse, if the revised Query Report is then transmitted to the clinical site using **DF\_QCfax** all of the queries DFdiscover thinks are in that report will be marked as sent, including any queries that were not written to the final version of the report. As a result the Query database may not contain the correct information regarding the most recent transmission of some queries. Then, if the `-o` option is used at a later date to specify that queries are not to be included in the report unless a specified period of time has elapsed since the queries were last transmitted, some queries may be omitted because they were mistakenly recorded as having been sent in the previous report.

- Once an external Query Report has been scheduled for transmission to the site using **DF\_QCfax** you should not create another external Query Report for that site until at least the next day. Here's why.

When a Query Report is scheduled for transmission using **DF\_QCfax** DFdiscover makes a copy of the current version of the report and hands it off to the fax or email software (depending on how you deliver external Query Reports to each site). This is the version that will be delivered to the site. Once an external Query Report has been transmitted DFdiscover will block attempts to create another version of that report on the same day, but until then any new versions you create will overwrite the existing version. Such new versions will not be sent unless you run **DF\_QCfax** again after creating them. But this does not replace the previous version and both will end up being transmitted to the site. You could schedule all Query Reports to go late at night and thus send many versions of the same Query Report to a site on the same day in this way.

In addition to confusing the clinical sites with more than one Query Report of the same name, the other negative consequence is that only one ASCII version of the report with the name `ccc-yyymmdd`, for each site and date, can live in the study `reports/QC/sent` directory, and it may not even be the one that was sent. It will simply be the ASCII version of the report that existed in the study `reports/QC` directory at the time the last version of the report to be successfully sent was being processed.

In general it is not a good idea to send external Query Reports to the clinical sites more than once a week, but in short studies external Query Reports might be sent daily. If it is necessary to send a second Query Report on the same day, it could be created using the internal Query Report option, and transmitted to the clinical site using **DF\_QCfax**, but such transmissions are not recorded and the status of queries transmitted in internal reports are not updated in query records.

- **Do not rely on queries for Query Report history.** When external Query Reports are created all queries which are written to the newly created reports are updated. The query status field is changed to `new`, `in report` and the fields which identify the location of the query in a report (by report date, page and item number on the page) are updated. When an external Query Report is sent to the clinical sites using **DF\_QCfax** the query status field is changed to `sent`. These changes overwrite any previous values that existed in these fields. Hence, each query holds information that can be used to identify the most recent external Query Report to which it was written, but any prior report history is lost. Query history can however be traced using report **DF\_ATmods**.
- **Use the `-c` and `-C` options carefully .** The `-c` and `-C` options may be used together in the same execution of **DF\_QCreports** to specify that a subset of subjects are desired for one or more of the specified study sites. However, it should be noted that if a site is specified using `-c` and `-C` is also used to specify only some of the subjects in the same



site, the `-c` option will be ignored because all subjects from the site have already been requested with the `-c` option. Thus, if a subset of subjects is desired, only the `-C` option should be used and the same site should not be requested with the `-c` option.

- **Use the `-o` option consistently.** The `-o` option is used to give the clinical sites a specified number of days to respond to the Query Report before including the queries in a new one. This works by checking the last Query Report date for all queries with status `sent`. However, if `DF_QCreports` has already been run without the `-o` option, the queries written to the newly created external Query Reports will have had their status reset to `new`, `in report` and thus the `-o` option will have no effect on these queries. To produce the desired effect, the `-o` option must be used consistently in the creation of `DF_QCreports`.
- **Use `-F` with `-P` and `-s` options carefully.** The queries included in the Query Report will represent the intersection of these specifications. For example, if the plate specifications in the `-P` and `-F` options do not intersect (`-P 2 -F 0:1:8-15`), no queries will be included in the Refax/Q&A section of the report.
- **Run `DF_QCsent` manually if Query Reports are not delivered using `DF_QCfax`.** When `DF_QCfax` is used to fax external Query Reports, it automatically runs `DF_QCsent` to update the Query database, marking all queries in faxed Query Reports as `sent`. However, if the Query Reports are distributed by some other means, this won't happen. As a result, `DFexplore` will incorrectly show some queries as `in report`, `not sent`, instead of `sent`. Additionally, if the `-o` option is used to give the clinical sites a chance to respond to a previous Query Report before including the query on a new one, `DFdiscover` won't know that these queries have been sent and thus will not wait the specified period before writing them into the next external Query Report. This limitation can be overcome by running `DF_QCsent` manually to register Query Reports which are successfully transmitted to the clinical sites.
- **Only internal Query Reports can be created when in read-only mode.** When the study server is in read-only mode, it will not be possible to make changes to the Query database, and will also, not be possible to execute `DF_QCupdate` or generate a full, 3-part external Query Report. It is possible, however, to generate an external Query Report containing only a cover page, messages, and/or the subject status summary as these do not modify the Query database in any way. Internal Query Reports can also be generated when the study database server is in read-only access mode.
- **Internal queries can only be included in internal reports.** Internal queries are not included in external reports, and there is no option that allows this.
- **Resolved queries can only be included in internal reports.** Resolved queries are not included in external reports, and there is no option that allows this.
- **Define cover sheets no longer than 1 page.** `DF_QCreports` expects that the cover sheet and all messages will fit on a single cover page for each site. It does not create automatic page breaks. Also, it does not perform line wrapping on report titles or cover sheets. Each report title, section sub-title, and text block is printed exactly as it appears in the specification file. Be careful when choosing fonts and using variable substitution to ensure that text lines do not become longer than expected.
- **Make sure file `QCmessages` is still relevant.** Messages will continue to go out on each report until the message file is changed. If you want to keep a message for use later on, but do not want to send it to anyone on the next batch of Query Reports, leave the site ID string blank, i.e. use

```
<FOR site="">
```

- **Remove unsent Query Reports.** External reports will remain in the study `QC` directory until they are registered as sent, at which point they are moved to the `QC/sent` directory. If a report is never registered as sent, the queries which it contained will be put into the next external Query Report created for that clinical site. The previously created and unsent Query Report then becomes irrelevant and should be removed from the `QC` directory. This is not done automatically and must be done manually by removing the file(s).

Failure to remove old external Query Reports may lead to unanticipated results if `DF_QCfax` is run with the `-a` option. This option requests the transmission of all unsent external Query Reports in the `QC` directory. Thus, if this directory contains a lot of old Query Reports, some clinical sites will receive a number of old and irrelevant Query Reports.

## See Also

[DF\\_QCupdate](#)  
[DF\\_QCfax](#)  
[DF\\_QCsent](#)

## Name

**DF\_QCsent** — Mark Query Reports as sent

## Synopsis

```
DF_QCsent {study} { [-s] | [-u] } [ report(s) ] [ -f name ]
```

## Description

**DF\_QCsent** is used to register successfully faxed Query Reports in the Query database. Invoked with `-s`, it sets the status of all queries included in the specified reports to "sent" and moves the reports to the `QC/sent` directory. Invoked with `-u`, it resets the status of all queries included in the specified reports to "unsent" and moves the reports from the `QC/sent` directory back to the `QC` directory. The latter option is commonly used to fix errors and undo a previous, incorrect `-s` invocation. As **DF\_QCsent** modifies the Query database by changing the status of queries, it will not be possible to execute **DF\_QCsent** when the study server is in read-only access mode.

If you use **DF\_QCfax** to fax Query Reports, it is not necessary to subsequently run **DF\_QCsent** as **DF\_QCfax** will automatically run **DF\_QCsent**, marking queries in all faxed Query Reports as sent. However, if you send out Query Reports manually (via paper fax, surface mail, etc), **DF\_QCsent** must be run manually to register Query Reports which are successfully transmitted to the study sites.

### Important

Use of **DF\_QCsent** (either automatically via **DF\_QCfax** or manually) is critical if `-o` is being used in **DF\_QCreports**. The option gives the clinical sites a chance to respond to a previous Query Report before reasking about a problem. If a successful fax is not registered DFdiscover assumes that it was not sent and thus all unresolved queries from that report will reappear on the next Query Report, regardless of how soon it is created.

In addition to **DF\_QCreports**, other DFdiscover applications depend on the registration of successfully transmitted documents. These include:

- **DFexplore.** The status of queries displayed by **DFexplore** distinguishes among 3 types of new queries: "not in any report", "in a report which has not been sent", and "sent". Queries will only be marked sent if their Query Reports are registered.
- **DF\_QCstatus.** The report status will not be correctly listed unless sent reports are registered.
- **DF\_QCprint.** **DF\_QCprint** requires the specification of report status (sent, unsent, internal).

## Registering most (but not all) newly created reports as successfully sent

This section offers two solutions to the problem of marking a large subset of the current set of new reports as being sent.

**Method 1.** This first solution requires some text editing skills.

1. Copy the `QC_NEW` file to a new file e.g., `temp`.

```
cp QC/QC_NEW temp
```

2. Edit `temp` to remove the 2 unsuccessful Query Reports.

```
vi temp
```

- Execute **DF\_QCsent** with `-f temp` to register the successful documents as sent.

```
-s -f temp
```

**Method 2.** An alternative solution is to first register all of the newly created reports followed by registering the 2 unsuccessful reports as unsent.

- Register all of the newly created reports as sent.

```
-s
```

- Register the 2 unsuccessful reports (011-990415 and 022-990415 in this example) as unsent.

```
-u 011-990415 022-990415
```

## Important

There is a drawback to this approach that may not be immediately apparent. Following these two steps, the audit trail would show that all of the queries in reports 011-990415 and 022-990415 were first marked as sent and then shortly thereafter marked as unsent. This has no effect on the database but may not be seen as a desirable part of the audit trail.

## Options

**-s-u** Register the selected Query Reports as sent or unsent, respectively (required).

Invoked with `-s`, the status of all queries included in the specified reports are set to "sent" and the reports are moved to the `QC/sent` directory. Invoked with `-u`, the status of all queries included in the specified reports are reset to "unsent" and reports from the `QC/sent` directory are moved back to the `QC` directory.

**report(s)-f name** Select the reports to be registered.

It is possible to use UNIX wild-card characters in the specification of reports but in this case the specification must be enclosed in quotes, like this `"*-980622"`.

The *name* specified in `-f name` may be an absolute or relative pathname. If it is relative, it is assumed to be relative to the study's `QC` directory.

If report numbers are specified, they are selected. If no reports are specified, and `-f name` is not present, all reports listed in `QC_NEW` are selected. If no Query Reports are specified, and `-f name` is present, all reports listed in *name* are selected. It is an error to specify reports and `-f name` in the same invocation. However this can be easily achieved by two invocations of **DF\_QCsent**.

## Examples

### Example 2.81. Register 2 specified Query Reports as sent

```
-s 023-910815 024-910815
```

### Example 2.82. Register as sent all of the unsent reports created on September 22, 2017

```
-s "*-170922"
```

## See Also

[DF\\_QCreports](#)  
[DF\\_QCfax](#)  
[DF\\_QCstatus](#)  
[DF\\_QCprint](#)

## Name

**DF\_QCstatus** — List existing Query Reports

## Synopsis

```
DF_QCstatus {study} [ [-c] | [-u filename/all] | [-s filename/all] | [-i filename/all] ]
```

## Description

**DF\_QCstatus** lists the current status and number of pages in existing Query Reports.

This report can be used to identify the names and sizes (in number of pages) of sent, unsent and internal Query Reports.

Specific reports can be selected by specifying the report identifier(s) and the report status where `-u` selects unsent reports, `-s` selects sent reports, and `-i` selects internal reports. If none of these options is specified, status is shown for all reports listed in `QC_NEW`, i.e. Query Reports created by the last invocation of **DF\_QCreports**.

The `-c` option produces a list of report names that are referenced by unresolved queries in the Query database. For each report name associated with one or more unresolved queries, the number of unresolved queries, missing pages and overdue visits is listed. Any of these reports which have not been registered as successfully faxed are marked with a `*` symbol. The number of new queries, not currently in any report is also shown.

## Options

- |                              |  |
|------------------------------|--|
| <code>-u filename all</code> | Select all, or specified, unsent Query Reports (from the <code>QC</code> directory)        |
| <code>-s filename all</code> | Select all, or specified, sent Query Reports (from the <code>QC/sent</code> directory)     |
| <code>-i filename all</code> | Select all, or specified, internal Query Reports (from <code>QC/internal</code> directory) |
| <code>-c</code>              | Select Query Reports referenced by unresolved queries                                      |

## Examples

### Example 2.83. List all unsent Query Reports

```
-u all
```

### Example 2.84. Determine if a specified report is registered as sent

```
-s 001-910815
```

### Example 2.85. List all Query Reports sent to site ID 001

```
-s "001-*
```

If the `*` symbol is used, the selection criteria must be enclosed in `"`.

### Example 2.86. List sent reports created on Aug 15,2011

```
-s "*110815"
```

**Example 2.87. List all internal Query Reports whose name begins with `stat.91`**

```
-i "stat.91*"
```

## Name

**DF\_QCupdate** — Update Query database

## Synopsis

```
DF_QCupdate {study} [ [-o] | [-d #, #-#] | [-s] | [-last] ] [-t yyyyymmdd]
```

## Description

**DF\_QCupdate** applies the visit scheduling and CRF page requirement rules that have been specified for the study.

**What does DF\_QCupdate do? .**

- **DF\_QCupdate** adds queries to identify overdue visits and missing CRF pages.
- **DF\_QCupdate** removes these same queries when they are no longer relevant, because the CRF pages have arrived in the database, a missed visit plate has arrived, the CRF plates have been registered as missed, or because the visit requirements have been changed in the study setup. **DF\_QCupdate** does not add or remove any other queries, i.e. those added by users or edit checks. Code 23 EC missing page queries added by the `dfaddmpqc()` edit check function are not removed or replaced by **DF\_QCupdate**.
- **DF\_QCupdate** updates existing overdue visit queries if there has been any change in the scheduled target date for the visit, or the visit label specified in the visit map.
- **DF\_QCupdate** creates a DFdiscover retrieval file listing any CRF pages which exist in the study database but that should not be there, e.g. a follow-up that was not expected because the subject died before it was due.
- **DF\_QCupdate** creates summary files, in the study work directory, that are read by other programs that need to know the current status of subject follow-up, including visits and CRF pages completed, scheduled, overdue, missing and unexpected.

**How does DF\_QCupdate know what visits and CRF pages are expected? .** Visit scheduling rules and CRF requirement rules are entered in the study setup tool where the dates that represent subject visits are identified using the VisitDate attribute, and input GUIs are provided for:

- a visit map which shows all study visits, their type (e.g. baseline, follow-up, termination, etc.), when they are due and what CRF plates are required and optional. The visit map allows visits to be grouped into one or more cycles or related visits, e.g. a screening cycle, 1 or more in-study treatment and follow-up cycles, and a cycle consisting of unscheduled event reports.
- a conditional visit map to identify visits that become required, optional or unexpected when a specified condition is met
- a conditional cycle map to identify visit cycles that become required, optional or unexpected when a specified condition is met
- a conditional plate map to identify CRF plates that become required, optional or unexpected when a specified condition is met
- a conditional termination map to identify visits at which follow-up is terminated, either just for the current visit cycle, or for all cycles, when a specified condition is met

A full description of these specifications and how they are used can be found in [Study Setup User Guide, Conditional Tests](#)

**How does DF\_QCupdate know whether expected visits and CRF pages have arrived? .** **DF\_QCupdate** launches **DF\_XXkeys** which exports the key fields and visit dates from all records in the study database having status `final`, `incomplete`, `lost` or `pending`. It also exports all of the trigger fields used by the conditional cycle, visit, plate and



termination maps, from these records. Data records with status `pending` are not exported or used by **DF\_QCupdate** in any way since the key fields on these data records are by definition unreliable. Thus `pending` records will never be counted among the records that have arrived in the database, or be used to trigger a condition. While it might be argued that records saved with `pending` in **DFexplore** should also be considered unreliable, if we ignored these records missing page and overdue visit queries would be created for records that are clearly underway, the visit dates on these records would not be available for scheduling the next visit, and conditions used to identify required cycles, visits and plates would not be triggered. While pending records might include some errors which have not yet been corrected this is also true of incomplete records. Thus on balance we have decided that it is better to include these records than to exclude them.

**When should I run DF\_QCupdate? .** **DF\_QCupdate** should be run before you run any of the programs that depend on the summary files it writes to the study work directory, or that depend on the Query database being up to date with regard to overdue visit and missing page queries. In some cases you may not care if the summary files and Query database have not been updated within the last few hours or days, and in other cases you may need them to be more current.

**Which programs and reports depend on DF\_QCupdate? .**

- **DF\_QCreports** is used to create the Query Reports sent to each participating clinical site, showing the first, last and next scheduled visits, and all unresolved data queries, including those for overdue visits and missing pages. It is particularly important that these reports be accurate, otherwise you risk mis-informing and upsetting the clinical sites. When creating external Query Reports, to be sent to the clinical sites, **DF\_QCreports** checks to see if **DF\_QCupdate** is running and stops with a warning message if it is. This ensures that these reports will not contain old information which is in the process of being updated.
- **DFexplore** reads the Query database and reports the number of queries of each type (including overdue visit and missing page queries).
- **DF\_PTvisits**, which is used to create reports showing visit scheduling information depends on summary files created by **DF\_QCupdate**.
- **DF\_CTqcs** creates reports showing the number and status of queries for each clinical site by reading the Query database
- **DF\_PTcrfs** lists the CRF pages received and missing for each visit. Missing CRF pages are determined by reading the Query database.

**Are there any restrictions on who can run DF\_QCupdate or when it can run? .**

- Only users with full permissions (e.g. no restrictions) can run **DF\_QCupdate**. The user must have full permissions on Data, Queries and Reasons at all levels (0-7).
- If you try to run **DF\_QCupdate** when it is already running you will receive an error message indicating that **DF\_QCupdate** is already in progress.
- **DF\_QCupdate** can be run while data entry is being performed. However, users should be aware that the update will only be accurate with regard to the state of the database when **DF\_QCupdate** was started.
- When the study server is in read-only mode, it is not possible to make changes to the Query database. **DF\_QCupdate** checks the study server access mode and exits with an abort message if the server is in read-only mode.

**What summary files does DF\_QCupdate create? .** In addition to updating the Query database with respect to overdue visit and missing page queries, **DF\_QCupdate** also creates a number of summary files, which are stored in the study work directory, and described in the programmers manual. As a first step **DF\_QCupdate** runs **DF\_XXkeys** to identify the visits that have been completed and the CRF pages that have been received for each subject. This step results in the creation of:

- `DFX_keys` - all required plates received at each visit for each subject
- `DFX_visit.dates` - all visit dates recorded for each subject
- `DFX_visit.date` - one visit date for each visit (preferred or available)

- VDillegal.drf - a DFdiscover retrieval file listing all illegal visit dates
- VDincon.drf - a DFdiscover retrieval file listing all inconsistent visit dates

Next, **DF\_QCupdate** reads the conditional maps and creates a summary file containing a record for each condition that was met for each subject:

- DFcvisit\_map - conditional visit map
- DFccycle\_map - conditional cycle map
- DFcplate\_map - conditional plate map
- DFcterm\_map - conditional termination map

**DF\_QCupdate** then compares the data exported from the study database to the files listed above with the study visit map rules and generates overdue visit and missing page queries which it imports into the study database. These queries are distinguished from the queries created by users by the category stored in each query (21 for missing page queries, and 22 for overdue visit queries). **DF\_QCupdate** will not create a code 21 query for a missing CRF page if a code 23 query (i.e. a missing page query created by an edit check using function dfaddmpqc) already exists in the study database. In addition to creating new queries **DF\_QCupdate** also deletes any old code 21 and 22 queries that are no longer relevant, and updates any old code 21 queries for which the plate label has been modified, and any old code 22 queries for which the visit label or visit date have been modified. In addition to creating, deleting and updating code 21 and 22 queries, **DF\_QCupdate** generates the following summary files which remain in the study work directory:

- DFX\_schedule - all completed and scheduled visits for each subject
- DFX\_studydates - all unique visit dates and their Julian values (days since Jan 1,1900)
- DF\_QCupdate.log - a summary of results from the last execution of **DF\_QCupdate**
- - a DFdiscover retrieval file listing unexpected CRFs received

**How are DF\_QCupdate and Schedule View related?** Schedule View, in **DFexplore**, is quite similar to **DF\_QCupdate** in its goals. Both report the visit schedule information for study subjects and identify missing pages and overdue visits. The schedule information presented by both is the same. However, the manner in which the information is generated and maintained differs in several important ways:

1. The results calculated by **DF\_QCupdate** are static - they are correct at the time that the report is run and then slowly age until refreshed when **DF\_QCupdate** is run again. The results in Schedule View are always up-to-date each time that the Schedule View is presented.
2. To reflect the results calculated by **DF\_QCupdate**, missing page and overdue visit queries are maintained in the study database. Schedule View does not create missing page or overdue visit queries.
3. **DF\_QCupdate** requires a user with full database permission to run the report, including the permission to add queries. The results are calculated for the entire study every time. Schedule View can be narrowed down to an individual site, or even an individual subject. The Schedule View does not need query write permission and can also be calculated from a read-only database.
4. The output from **DF\_QCupdate** is easily shared with external parties, who have no access to the study database, by emailing a PDF generated by **DF\_QCreports**. Schedule View requires that the user has access to **DFexplore** and at least read access to their site or subject data.

## Options

- o do not update overdue visit queries (update missing page queries only)

- d #, #-#** delete all queries with the specified query category. If no categories are specified, delete all missing page (code 21) and overdue visit (code 22) queries.
- s** when a subject terminates count visits overdue if scheduled ON or BEFORE the termination date. Without this option visits are overdue only if scheduled before the termination date.
- last** show date and results from last execution of **DF\_QCupdate**, then quit
- t yyyymmdd** set today's date for visit scheduling and overdue visit calculations

## Examples

### Example 2.88. Show results from last execution of DF\_QCupdate then quit

```
-last
DF_QCupdate - jane Mon Mar 29 16:02:38 2018
today = 20180329 38074
Missing Pages      Update =   10    Total =   44
Overdue Visits     Update =   15    Total =  126
Unexpected Records Update =    0    Total =    9
```

This example shows that **DF\_QCupdate** was last executed by jane on Mar 29, 2018 at 38 seconds after 16:02 (4:02 pm). The value, 38074 shown on the next line is the Julian value of the date (calculated as the number of days since Jan 1, 1900), and is the value used to determine how many days visits are overdue. The Update values on the next 3 lines show the number of records that need to be added, modified or deleted, for each category and the Total values show the resulting total for each category that will exist after the updates are applied.

Occasionally, as illustrated in the next example, some updates may become irrelevant before they can be applied.

### Example 2.89. Update the Query database and all summary files

```
no options
DF_QCupdate: Update Query Database.      DFstudy 253. Mar 31,2018 16:55

1. Checking permissions on study 253, Demo Study 253
2. Reading study configuration files
3. Exporting current overdue visit and missing plate queries
4. Exporting required data from study database:
   -exporting visit dates and keys from required plates
   -exporting data for conditional terminations
   -exporting data for conditional plates
5. Checking for overdue visits, missing plates and unexpected records
Missing Pages      Update =    8    Total =   48
Overdue Visits     Update =    9    Total =  129
Unexpected Records Update =    0    Total =    9
6. Updating the Query database (using DFimport.rpc)
   adding new queries, replacing modified queries, deleting queries no longer relevant
   E** line 9 failed 'Import failed as existing record is in use'
   Visit 21 is no longer overdue for ID 1023
   17 records input: 16 imported (0 warnings), 1 failed with errors
7. Done
```

This example shows the output messages displayed by **DF\_QCupdate** as it proceeds step by step through the current study scheduling specifications and existing subject data. In step 6 the required updates for missing page and overdue visit queries

are applied to the Query database. If users are working on the study while **DF\_QCupdate** is running it is possible that a missing page or overdue visit identified during step 5 will have been entered before step 6 begins. When this occurs a 2 line message is written for each update record that is rejected. The final line of step 6 notes the total number of queries submitted for update, the number that were applied, and the number that failed. In the above example one of the overdue visit queries was rejected because the visit was no longer overdue when the updates were submitted to the database server in step 6. As a result the total number of overdue visit queries in the database will be 128 (1 less than the number calculated in step 5).

### Example 2.90. Delete all overdue visit and missing page queries

```
-d
DF_QCupdate: Update Query Database.      DFstudy 252. May 20,2017 10:33

1. Checking permissions on study 252, demo252
2. Reading study configuration files
3. Exporting current overdue visit and missing plate queries
4. Discarding missing plate and overdue visit queries
   12 records input: 12 imported (0 warnings), 0 failed with errors
   Discarded 5 missing page queries
   Discarded 7 overdue visit queries
5. Done
```

When the `-d` option is used all current overdue visit and missing page queries are deleted from the Query database. This is all that is done. No subject scheduling is performed, no new overdue visit or missing page queries are created, and the summary files stored in the study work directory are not updated.

The number of records described as input in step 4, is the number of queries submitted to the study server for deletion. The total (12 in the above example) should match the total number of missing page and overdue visit queries deleted as reported in the final 2 lines of the output (5 + 7 in the above example).

### Example 2.91. Delete missing page queries created by DF\_QCupdate and dfaddmpqc

```
-d 21,23
DF_QCupdate: Update Query Database.      DFstudy 252. May 20,2017 09:57

1. Checking permissions on study 252, demo252
2. Reading study configuration files
3. Exporting current overdue visit and missing plate queries
4. Discarding all queries with specified categories
   Category 21 :      14 queries
   Category 23 :       2 queries
   16 records input: 16 imported (0 warnings), 0 failed with errors
5. Done
```

Each query has a category. The standard categories are: 1=missing, 2=illegal, 3=inconsistent, 4=illegible, 5=fax noise, 6=other problem, 21=missing page, 22=overdue visit, and 23=EC missing page (queries created by edit check function `dfaddmpqc`). Codes ranging from 30-99 may also exist if other query categories have been defined at the study level. When the `-d` option is followed by a list of query categories, the queries with the specified categories are deleted. No subject scheduling is performed, no new overdue visit or missing page queries are created, and the summary files stored in the study work directory are not updated.

#### Note

Query categories 21 and 22 can be re-created by running **DF\_QCupdate**, and code 23 queries can be re-created using an appropriate batch edit check run, but there is no automatic way to re-create queries that were added

manually by individuals using **DFexplore**. There is no undo feature. Deletes are permanent, and the only record of their prior existence will be in the audit trail. Use the `-d` option with care.

## See Also

[DF\\_XXkeys](#)  
[DF\\_QCreports](#)  
[DF\\_SSvisitmap](#)  
[DF\\_ICvisitmap](#)

## Name

**DF\_QCview** — Display requested Query Reports

## Synopsis

```
DF_QCview {study} [ [-u #] | [-s #] | [-i name] ]
```

## Description

**DF\_QCview** displays the contents of one or more Query Reports. The formatting which is performed prior to faxing and printing is not performed by **DF\_QCview**. **DF\_QCview** provides a convenient way to quickly check the contents of a Query Report before it is printed or faxed. To see exactly what a report will look like when the investigator receives it, use **DF\_QCprint**.

If no options are used, the reports listed in `QC_NEW` (that is, those reports created by the last execution of **DF\_QCreports**) are output.

## Options

- u #**        Select the specified unsent Query Report
- s #**        Select the specified sent Query Report
- i name**    Select the named internal Query Report

## Examples

### Example 2.92. Display the specified unsent Query Report

```
-u 001-951002
```

### Example 2.93. Display the specified sent Query Report

```
-s 001-910815
```

### Example 2.94. Display the specified internal Query Report

```
-i status951001
```

## See Also

[DF\\_QCprint](#)  
[DF\\_QCfax](#)

## Name

DF\_SScenters — Display study site information

## Synopsis

DF\_SScenters {study} [-s]

## Description

Display for each site, the following information:

CID <sup>❶</sup>	NAME <sup>❷</sup>	CLINICAL SITE <sup>❸</sup>	PHONE <sup>❹</sup>	FAX <sup>❺</sup>
------------------	-------------------	----------------------------	--------------------	------------------

- ❶ the site ID
- ❷ the name of the contact person
- ❸ the name of the study site
- ❹ the contact person's phone number
- ❺ the primary fax number (or email address)

By default, no sorting is performed and sites are simply listed in the order that they were defined.

## Options

-s Sort the output by increasing site ID

## Example

### Example 2.95. Listing for a sites database with 3 sites

```
DF_SScenters: List Study Sites Database.           DFstudy 254. Nov 21,2015 20:21
CID NAME                CLINICAL SITE                PHONE                FAX
010 Person #1           Hospital #1                   1-888-123-4567      1-888-765-4321
020 Person #2           Hospital #2                   1-866-234-5678      1-866-876-5432
099 Error Monitor       DF/Net Research, Inc.        1-877-345-6789
mailto:c099@dfnetresearch.com
```

## Name

**DF\_SSschema** — Display detailed data dictionary information

## Synopsis

```
DF_SSschema {study} [-f] [-p # #-#] [-e] [-s] [-v string] [-V string] [-D string] [-T string] [-M string] [-m string]
```

## Description

**DF\_SSschema** displays a detailed description of variables selected from the study data dictionary. The detail for each variable includes:

```
Plate  Field(*|+|-)①(Ⓜ)② Name           Description③           <Alias Name>④
                                Module[Module Instance] FieldName
                                Type Style Length Format Legal Values⑤
                                Pivot Imputation ⑥
                                Coding           ⑦
                                Skip              ⑧
                                Edit checks      ⑨
```

- ① If the field is:
  - essential, a \* is displayed after the field number,
  - required, a + is displayed after the field number,
  - otherwise, a - is displayed, or nothing.
- ② If a minimum reason level is defined, the reason level is displayed.
- ③ If the description is longer than 25 characters (the length printed on Query Reports), the description is followed by the length and a ?.
- ④ If the variable's alias name is different from the name, the alias name is displayed at the end of the line enclosed in < and >.
- ⑤ If format and legal values are both undefined then field length is printed, otherwise the format and/or legal values are printed. Note that if legal range definition for the subject ID field includes the meta-word \$(ids), the literal \$(ids) will be reported as the legal value rather than the (lengthy) concatenated list of subject IDs from all sites.
- ⑥ The pivot year and imputation method are displayed only if the variable type is date. The pivot year appears only for dates which are defined with a 2 digit year. The default pivot year (if none is specified) is 1940. The date imputation method must be one of: none, start of month and/or year, middle of month and/or year, or, end of month and/or year.
- ⑦ Coding information appears only for those variables that define coding information.
- ⑧ Skip information appears only for those variables that define it.
- ⑨ Edit checks appear if they are defined for the variable. The edit check name appears together with the invocation method which is one of plate enter, field enter, field exit, or plate exit.

The options `-v`, `-V`, `-D`, and `-T` each accept only one string. If any string includes spaces, it must be enclosed in ". String matching is case sensitive.

When selection options are used in combination, selected variables can match on any of the selection criteria.

The order of plates in the output is the same as the order that plates appear in the schema and in **DFsetup**. The description of each plate includes information from the **Plate > Edit** dialog in **DFsetup**, and includes the plate number, label, and whether or not there is a Preprocess or Postprocess procedure.



## Options

<b>-f</b>	Start the output with a description of all study plates
<b>-p # #-#</b>	Select variables only from the specified plates
<b>-e</b>	Select variables that have edit checks defined
<b>-s</b>	Select variables that have skip patterns defined
<b>-v string</b>	Select variables with string in the variable name
<b>-V string</b>	Select variables with string in the alias variable name
<b>-D string</b>	Select variables with string in the variable description
<b>-T string</b>	Select variables with string in the variable type or style
<b>-m string</b>	Select variables with string anywhere in the variable definition. If more than one string is specified, select cases if ANY of the strings are matched.
<b>-M string</b>	Select variables with string anywhere in the variable definition. If more than one string is specified, select cases if ALL of the strings are matched.

## Examples

**Example 2.96. Display data dictionary information for all variables on plates 7, 8 to 10 inclusive, and 22**

```
-p 7 8~10 22
```

**Example 2.97. Display the description of all study plates, plus all fields on plates 1, 2 and 3**

```
-f -p 1~3
```

**Example 2.98. Display data dictionary information for all variable(s) with DRUG in the variable name**

```
-v DRUG
```

**Example 2.99. Select and display all variable(s) with sore throat in the variable description**

```
-D 'sore throat'
```

**Example 2.100. List variables defined using the Visit Date attribute**

```
-T VisitDate
```

**Example 2.101. Display data dictionary information for all variables with edit checks on plates 1, 2 and 3.**

```
-p 1~3 -e
```

## Name

**DF\_SSvars** — Display essential variable information from data dictionary

## Synopsis

```
DF_SSvars {study} [-S parts format] [-P #, #-#] [-F #, #-#] [-prune] [ [-M string] | [-m string] ] [-v]
```

## Description

**DF\_SSvars** lists variables defined in the study data dictionary.

The report creates a compact variable listing of selected information from the data dictionary. The presentation is typically one line per variable, unlike [DF\\_SSschema](#), which presents multiple lines per variable.

The output is organized by plate, where each plate begins with plate-specific information followed by the variable listing.

The plate-specific information includes:

1. plate number
2. plate descriptive label
3. is `DFSEQ` in the barcode or the first data field on the page?
4. TYPE information from the visit map, file map and conditional plate map where the value may be one or more of:
  - **Required.** listed as required for at least one visit
  - **Optional.** listed as optional for at least one visit
  - **Conditional.** defined in the conditional plate map
  - **Missed Visit Notification.** plate signals that the visit was missed
  - **Early Termination.** plate signals early termination of a visit cycle

The variable-specific information for each plate lists each variable, one per line. Each line begins with the variable field number <sup>1</sup>. If the field is:

- essential, a \* is displayed after the field number,
- required, a + is displayed after the field number,
- otherwise, a - is displayed, or nothing.

If a minimum reason level is defined, the optional/required/essential symbol is followed by the minimum reason level.

Any combination and ordering of the following parts of the variable definition may be selected using the -S option.

**v** variable name

**V** alias

---

<sup>1</sup>The variable field number also corresponds to the order in which variables were defined on the plate, the order in which they are traversed in **DFexplore** and the order in which they are stored in the study database records.

<b>D</b>	variable description
<b>S</b>	style
<b>L</b>	legal values
<b>F</b>	format
<b>d</b>	VisitDate tag (identifies subject scheduling visit dates)
<b>P</b>	pivot year for dates
<b>i</b>	imputation method for dates
<b>H</b>	help message
<b>W</b>	maximum width (stored in database)
<b>w</b>	display width (visible on-screen)
<b>P</b>	plate number
<b>m</b>	module[instance on plate]
<b>M</b>	module description
<b>J</b>	edit checks on field entry
<b>K</b>	edit checks on field exit
<b>j</b>	edit checks on plate entry
<b>k</b>	edit checks on plate exit
<b>C</b>	code labels
<b>s</b>	skip field specifications
<b>T</b>	data type (int, string, check, choice, date, time)
<b>B</b>	blank space

Note: If a field attribute is too long to fit in the specified space it is truncated. In the format string %-25s means left justification in a 25 character space, and %25s means right justification. Additional space may be added between columns, e.g. -S v,D,S "%-15s %-35s %-25s" puts 2 spaces between each of the 3 columns.

Note that the value for variable type (code letter **T**) encodes both the variable's base type and it's style name. This can make it more difficult to retrieve variables by their style name only as the base type must also be known.

To include variable attributes other than those included by default, or to change the formatting used to display the attributes, use **-S**. The attributes to be included are specified by including their corresponding code letter(s) from the above list, comma-delimiting multiple code letters as necessary. The code letters must be followed by a formatting specification. If more than one code letter is included, the corresponding formatting specification must be enclosed in " as it will contain more than one formatting option. The attribute selected by the first code letter is formatted with the first formatting option, the second code letter with the second option, and so on. Formatting options follow the same specifications that can be found described in other sections, such as [Formatting Specifications](#).

To select the variables to be listed, use one or more of the **-P**, **-F**, and **-M|m** options. Use **-P** to select variables from specified plates, **-F** to select variables by field number, and **-M|m** to select variables which match on one or more specified strings.

String matching, as requested by `-M|m`, is performed case sensitive and independent of word boundaries. For example, `-M Date` will match `VisitDate` but not `date`. The match can occur on any variable attribute, not just those variable attributes selected by `-S`.

Matching can occur on more than one string. In such a case, the possible string matches are listed, space-delimited, following the option. If the variable to be selected must match on all of the specified strings, use `-M`. Conversely, if the variable to be selected may match on any of the specified strings (at least one must match), use `-m`.

## Options

<b>-S parts format</b>	Select variable attributes to be listed and their output format (default is: <code>v,D,S,W,F,L "%-9s %-25s %-15s %-5s %-8s %s"</code> )
<b>-P #, #-#</b>	Select plates
<b>-F #, #-#</b>	Select variables by their field position
<b>-prune</b>	Exclude variable if all requested attributes are not defined
<b>-M string-m string</b>	Select variable if all (or any in the case of <code>-m</code> ) specified strings are found in the definition
<b>-v</b>	Suppress the variable listing, showing only plate information

## Examples

### Example 2.102. Select variables from plate 2 containing the string date in any attribute

```
-P 2 -M date
DF_SSvars: Variable Specifications. DFstudy 2. Dec 09,2014 14:06          PAGE 1

PLATE 002 Patient Entry Form (23 fields)
      DFSEQ: in the barcode,  TYPE: Required

      Name      Field Description      Style      Store Format  Legal
  9-  VISDAT    Entry Date          S03 Study Date  8    dd/mm/yy 01/ ...
 12+ BDATE      Date of Birth (dd/mm/yy) SimpleDate    8    dd/mm/yy
 20+ SFDATE     Date of First Study Follo S60 Future Date 8    dd/mm/yy 01/ ...
```

Note that the match string, `date`, does not appear in the report output. In this case it matches the variable's type attribute which is not displayed by default.

### Example 2.103. List variables from plate 2 that have a variable type attribute beginning with date

```
-P 2 -M "%T date"
DF_SSvars: Variable Specifications. DFstudy 2. Dec 09,2014 14:08          PAGE 1

PLATE 002 Patient Entry Form (23 fields)
      DFSEQ: in the barcode,  TYPE: Required

      Name      Field Description      Style      Store Format  Legal
  9-  VISDAT    Entry Date          S03 Study Date  8    dd/mm/yy 01/ ...
```

```

12+  BDATE      Date of Birth (dd/mm/yy)  SimpleDate      8      dd/mm/yy
20+  SFDATE      Date of First Study Follo S60 Future Date 8      dd/mm/yy 01/ ...
    
```

The output in this case is the same as the output from the previous example. The data type of each of these variables is date although the variables have different variable style attributes.

### Example 2.104. List all required date variables from plates 1 and 2

```

-P 1,2 -M "%T date" "%A required"
DF_SSvars: Variable Specifications. DFstudy 2. Dec 09,2014 14:10          PAGE 1

PLATE 001 Blood Pressure Screening Visits (38 fields)
      DFSEQ: in the barcode,  TYPE: Required

      Name      Field Description      Style      Store Format      Legal
33+  BDATE      Date of Birth (dd/mm/yy)  SimpleDate      8      yy/mm/dd

PLATE 002 Patient Entry Form (23 fields)
      DFSEQ: in the barcode,  TYPE: Required

      Name      Field Description      Style      Store Format      Legal
12+  BDATE      Date of Birth (dd/mm/yy)  SimpleDate      8      dd/mm/yy
20+  SFDATE      Date of First Study Follo S60 Future Date 8      dd/mm/yy 01/ ...
    
```

### Example 2.105. List all dates and all required variables from plates 1 and 2

```
-P 1,2 -m "%T date" "%A required"
```

This example differs from the previous one in the use of `-m` instead of `-M`. This includes variables that match at least one of the selection strings. The resulting variable list is much larger than for the previous example and is omitted for that reason.

### Example 2.106. List the first 7 variables from all plates

```
-F 1~7
```

### Example 2.107. Display the name, description, and legal values for all variables on plates 1, 3 and 6 through 8 that have the string `visitDate` in an attribute

```
-S v,D,L "%-15s %-25s %s" -P 1,3,6~8 -M VisitDate
```

## See Also

[DF\\_SSschema](#)

## Name

**DF\_SSvisitmap** — List visit scheduling specifications for a study

## Synopsis

```
DF_SSvisitmap {study} [-b] [-i { [[vmap] | [visits] ] [ [vptbl] | [pvtbl] ] [ [vdates] | [etp] | [cterm] | [cplate] | [cvisit] | [ccycle] ... ] } ]
```

## Description

This report lists the subject scheduling specifications for a study:

1. **Visit Map.** The visit map defines all cycles and visits, due dates and overdue allowance, and the CRF plates for each visit (see [Study Setup User Guide, Subject Visit Scheduling](#)).
2. **Visit Dates.** Data fields defined with the VisitDate attribute identify those database fields used to record the date on which a subject assessment occurred (see [Study Setup User Guide, Visit Dates](#)).
3. **Early Termination Plates.** These plates correspond to CRF forms used by the clinical sites to report the early termination of a visit map cycle for a subject. They must be registered in the Plate map as described in [Study Setup User Guide, PlatesPlates](#).
4. **Conditional Termination Map.** These specifications identify database conditions that trigger early termination of a visit map cycle, or termination of all cycles (see [Study Setup User Guide, Conditional Tests](#)).
5. **Conditional Plate Map.** These specifications identify database conditions that modify the requirements for specified CRF plates at specified visits (see [Study Setup User Guide, Conditional Tests](#)).
6. **Conditional Visit Map.** These specifications identify database conditions that modify the requirements for the visits defined in the visit map (see [Study Setup User Guide, Conditional Tests](#)).
7. **Conditional Cycle Map.** These specifications identify database conditions that modify the requirements for the cycles defined in the visit map (see [Study Setup User Guide, Conditional Tests](#)).

## Options

When **DF\_SSvisitmap** is run with no options, all of the above specifications are included in a report which is printed as one long list of specifications.

- |   |  |
|---|--|
| <b>-b</b>   | Start each section of the report on a new page.  |
| <b>-i vmap visits vptbl pvtbl vdates etp cterm cplate cvisit ccycle</b> | Select any combination of the subject scheduling specifications to be included in the report. The output created by each keyword is illustrated in the examples. |

### Note

Only the first 3 characters of each keyword are significant. Thus `-i vmap` and `-i vma` produce the same result.

The order in which keywords are specified is irrelevant and has no effect on the order in which sections are included in the report. This output ordering is static.

## Examples

### Example 2.108. `-i vmap` option

```
DF_SSvisitmap: Visit Map Specifications. DFstudy 253. Mar 18,2018 10:54
```

```

Cycle 0: SCREENING - Screening
  91: X    0( 0) Screening Visit #1           (VisitDate 1.13)
        Req:1 Opt:102 MVP:none
  92: X    4( 2) Screening Visit #2           (VisitDate 1.13)
        Req:1 Opt:102 MVP:12
  93: X    8( 0) Screening Visit #3           (VisitDate 1.13)
        Req:1 Opt:102 MVP:12

Cycle 1: IN-STUDY VISITS - Required
  starts on day 2( 0) from termination of previous cycle
  0: B    0( 0) Baseline                       (VisitDate 2.9)
        Req:2-4 Opt:102 MVP:none
  21: S   30( 3) 1 Month Follow-up             (VisitDate 5.9)
        Req:5 Opt:15,102 MVP:12
  22: S   60( 5) 2 Month Follow-up             (VisitDate 5.9)
        Req:5 Opt:15,102 MVP:12
  23: S   90( 5) 3 Month Follow-up             (VisitDate 5.9)
        Req:5 Opt:15,102 MVP:12
  24: S  120( 5) 4 Month Follow-up             (VisitDate 5.9)
        Req:5 Opt:15,102 MVP:12
  30: T  120( 5) Study Termination             (VisitDate 7.9)
        Req:7-8 Opt:15-18 MVP:12

END OF ALL CYCLES
  40: A                Death Report           (VisitDate 10.9)
        Req:10 Opt:none MVP:none
  101-199: O           AE Report ##{S.2.2}
        Req:44 Opt:45 MVP:none
  
```

This option provides a complete listing of the specifications for all cycles and visits defined in the study visit map. If cycle definition records exist in the visit map the visits within each cycle will be listed below a cycle description showing the: cycle number (a sequential number starting at 0 for screening and 1 for the first in-study cycle), cycle label (a descriptive label used on reports like this one), cycle type (Screening, or for in-study cycles: Required, Optional or Conditional), and cycle scheduling (start day and overdue allowance, if the cycle has a scheduling method). The description of each visit begins with the visit number followed by a single letter visit type acronym (one of XPBSTWFEARrO), visit due day (the number of days following the cycle baseline visit), overdue allowance (after which the visit will be classified as overdue if not received), visit label (as it will appear on Quality Control reports to identify overdue visits), database field that contains the visit date (shown as VisitDate plate#.field#), and a list of required plates (Req), optional plates (Opt) and the missed visit plate (MVP).

### Example 2.109. -i visits option

```

DF_SSvisitmap: Visit Map Specifications.           DFstudy 253. Mar 18,2018 10:56

Cycle 0: SCREENING - Screening
  91: X    0( 0) Screening Visit #1
  92: X    4( 2) Screening Visit #2
  93: X    8( 0) Screening Visit #3

Cycle 1: IN-STUDY VISITS - Required
  starts on day 2( 0) from termination of previous cycle
  0: B    0( 0) Baseline
  21: S   30( 3) 1 Month Follow-up
  22: S   60( 5) 2 Month Follow-up
  
```



```

    23: S    90( 5)  3 Month Follow-up
    24: S   120( 5)  4 Month Follow-up
    30: T   120( 5)  Study Termination

END OF ALL CYCLES
    40: A                Death Report
101-199: O              AE Report #{S.2.2}
    
```

This option displays the visit map without the VisitDate and plate specifications, and is useful if you want to focus on visit scheduling alone.

### Example 2.110. -i vptbl option

```
DF_SSvisitmap: Visit Map Specifications.           DFstudy 253. Mar 18,2018 10:57
```

Visit Map Table of Visits (rows) by Plates (columns)

	1	2	3	4	5	7	8	10	15	16	17	18	44	45	102
91:	X	.	.	.	.	.	.	.	.	.	.	.	.	.	O
92:	X	.	.	.	.	.	.	.	.	.	.	.	.	.	O
93:	X	.	.	.	.	.	.	.	.	.	.	.	.	.	O
0:	.	X	X	X	.	.	.	.	.	.	.	.	.	.	O
21:	.	.	.	.	X	.	.	.	O	.	.	.	.	.	O
22:	.	.	.	.	X	.	.	.	O	.	.	.	.	.	O
23:	.	.	.	.	X	.	.	.	O	.	.	.	.	.	O
24:	.	.	.	.	X	.	.	.	O	.	.	.	.	.	O
30:	.	.	.	.	.	X	X	.	O	O	O	O	.	.	.
40:	.	.	.	.	.	.	.	X	.	.	.	.	.	.	.
101:	.	.	.	.	.	.	.	.	.	.	.	.	X	O	.

In visit by plate tables required plates are indicated by the letter 'X' and optional plates are indicated by the letter 'O'. Any visit lists defined in the visit map are represented by the first visit in the list. This is the case for the adverse event reports in our example, which have visit numbers 101-199. This option does not attempt to break or wrap tables into printable chunks and thus may produce results which are best viewed on screen.

### Example 2.111. -i pvtbl option

```
DF_SSvisitmap: Visit Map Specifications.           DFstudy 253. Mar 18,2018 10:58
```

Visit Map Table of Plates (rows) by Visits (columns)

	91	92	93	0	21	22	23	24	30	40	101
1:	X	X	X	.	.	.	.	.	.	.	.
2:	.	.	.	X	.	.	.	.	.	.	.
3:	.	.	.	X	.	.	.	.	.	.	.
4:	.	.	.	X	.	.	.	.	.	.	.
5:	.	.	.	.	X	X	X	X	.	.	.
7:	.	.	.	.	.	.	.	.	X	.	.
8:	.	.	.	.	.	.	.	.	X	.	.
10:	.	.	.	.	.	.	.	.	.	X	.
15:	.	.	.	.	O	O	O	O	O	.	.
16:	.	.	.	.	.	.	.	.	O	.	.
17:	.	.	.	.	.	.	.	.	O	.	.
18:	.	.	.	.	.	.	.	.	O	.	.

```

44: . . . . . . . . . . X
45: . . . . . . . . . . 0
102: 0 0 0 0 0 0 0 0 0 . . .
    
```

This option displays the same information as the previous example, but with columns and rows reversed.

### Example 2.112. -i vdates option

```

DF_SSvisitmap: Visit Map Specifications.           DFstudy 253. Mar 18,2018 10:59
All VisitDates Defined In The Study Schema:
001  13. S1DATE           Screen 1 Date
      date VisitDate F=dd/mm/yy L=01/01/01-today
      Pivot= 1940 Imputation= none
002   9. EDATE           Entry Date
      date VisitDate F=dd/mm/yy L=01/01/01-today
      Pivot= 1940 Imputation= none
005   9. FDATE           Visit Date
      date VisitDate F=dd/mm/yy L=01/01/01-today
      Pivot= 1940 Imputation= none
007   9. TDATE           Final Visit Date
      date VisitDate F=dd/mm/yy L=01/01/01-today
      Pivot= 1940 Imputation= none
010   9. DDATE           Date of Death
      date VisitDate F=dd/mm/yy L=01/01/01-today
      Pivot= 1940 Imputation= none
102   9. ETPtermdate     Termination Visit Date
      date VisitDate F=dd/mm/yy L=01/01/01-today
      Pivot= 1940 Imputation= none
    
```

This option shows the location of all database fields defined using the VisitDate attribute. Each field is identified by plate number followed by field number, name and label on the first line. The second line shows that these are date fields using the VisitDate attribute and identifies the format and legal value specifications. The 3rd line shows the pivot year and imputation method for partial dates. For an explanation of date specifications see Visit Dates in subject scheduling chapter of the study planning guide.

### Example 2.113. -i etp option

```

DF_SSvisitmap: Visit Map Specifications.           DFstudy 253. Mar 18,2018 10:59
Plates Which Signal Early Termination of the Current Cycle:
Plate 102 Early Termination Report
    
```

Early termination plates are listed by plate number and plate label. In this example the study has a single form which is used to indicate that the current cycle terminated early.

### Example 2.114. -i cterm option

```

DF_SSvisitmap: Visit Map Specifications.           DFstudy 253. Mar 18,2018 10:59
    
```

**Conditional Termination Map:**

1. IF Visit 0 Plate 4 Field 15 equals 1  
    AND Visit 0 Plate 4 Field 17 is less than 3  
    the cycle terminates at visit 0
  
2. IF Visit 91 Plate 102 Field 34 equals 5  
    all follow-up terminates at visit 91

This example illustrates the 2 possible types of conditional termination. The first specification terminates the cycle in which visit 0 is defined when both of the specified database conditions are met. In our example this is cycle 1. The second specification shows a condition which aborts all cycles based on a condition at visit 91, which in our example is the first visit of the screening cycle. As a result all follow-up will end at visit 91, and neither the remaining visits of cycle 0 nor cycle 1 will not be expected for any subject that meets this condition.

**Example 2.115. -i cplate option**

DF\_SSvisitmap: Visit Map Specifications. DFstudy 253. Mar 18,2018 11:00

**Conditional Plate Map:**

1. IF Visit 30 Plate 7 Field 18 equals 2  
    at visit 30 required plates: 15-18
  
2. IF at any Visit Plate 44 Field 32 equals 3,5,6,9  
    at visits meeting this condition required plates: 45
  
3. IF at any Visit Plate 44 Field 32 does not equal 3,5,6,9  
    at visits meeting this condition excluded plates: 45

This example shows 3 specifications. If the first condition is met, plates 15-18, which are defined as optional for visit 30 in the visit map (see example 1 above), become required at visit 30. The 2nd and 3rd conditions are related. Plate 44 is required for adverse event reports and plate 45 (defined as optional in the visit map) becomes either required or excluded (i.e. unexpected) depending on the values recorded in field 32 of plate 44 for each and every adverse event report.

**Example 2.116. -i cvisit option**

DF\_SSvisitmap: Visit Map Specifications. DFstudy 253. Mar 18,2018 11:00

**Conditional Visit Map:**

1. IF Visit 91 Plate 1 Field 12 is less than 80  
    excluded visits: 92
  
2. IF at any of Visits 102-199 Plate 44 Field 6 is greater than 0  
    required visits: 101-value (value = IF condition data field value)

In this example the first specification indicates that screening visit 92 is not expected if field 12 is less than 80 at screening visit 91. The second example shows a condition that checks for all adverse events having a lower report number, on the assumption that adverse event report numbers are assigned sequentially, and appear as assessment numbers 101 to 199.

**Example 2.117. -i ccycle option**

DF\_SSvisitmap: Visit Map Specifications. DFstudy 253. Mar 18,2018 11:01

**Conditional Cycle Map:**

1. IF Visit 0 Plate 2 Field 25 equals 1,3,4  
excluded cycles: 0
2. IF Visit 93 Plate 1 Field 33 equals 2  
excluded cycles: 1

In this example the first specification indicates that cycle 0 (the screening cycle) is not expected if a data value collected at the baseline visit of cycle 1 is equal to 1, 3 or 4. The 2nd specification indicates that cycle 1 is unexpected if a condition is met at visit 93, the last visit of the screening cycle.

## See Also

[DF\\_QCupdate](#)

[DF\\_ICvisitmap](#)

## Name

**DF\_WFcrfs** — Summary of daily journal transactions for CRF validation

## Synopsis

```
DF_WFcrfs {study} {-u userlist} { [-t yy/mm/dd-yy/mm/dd] | [-t ] } [-p #, #-#] [-x #, #-#] [-v #, #-#] [-f #] [-days]
[-status] [-test]
```

## Description

The report output shows the number of CRFs validated to each validation level on each day during the specified time period for the specified user(s). It also shows the hours during which the validation occurred and the amount of time required to perform the validations at each level.

Both `-u` and `-t` are required. A separate report is created for each user when multiple users are specified.

The report consists of one line per day, for which there was activity, during the specified date range. Each line shows the number of CRFs validated at each validation level that day. Totals appear at the bottom of the report along with statistics on the amount of time that the user spent validating CRFs at each validation level. If the specified date range spans more than one month, a separate report is created for each month.

The columns of the report appear under the common heading:

		PRIMARY RECORD VALIDATIONS PERFORMED <sup>❶</sup>							FINAL STATUS OF ALL CRFS <sup>❷</sup>					
DATE	FROM TO	V1	V2	V3	V4	V5	V6	V7	TOTAL	PRf	PRi	PRp	SEC	DEL
TOTAL <sup>❸</sup>														

- ❶ In the columns under PRIMARY RECORD VALIDATIONS PERFORMED each CRF is counted once per day at each validation level, V1 through V7. As a result, some CRFs may be counted more than once on the same day if they were validated to more than one level on that same day.
- ❷ In the columns under FINAL STATUS OF ALL CRFS each CRF page is counted only once per day and is counted at the record status last attained on that day.
- ❸ The final record status is categorized by:
  - PRf= primary final
  - PRi= primary incomplete
  - PRp= primary pending
  - SEC= secondary
  - DEL= deleted
  - and TOTAL which is the sum of the record statuses shown in that row.

There are 3 different totals shown at the bottom of each validation level column. The first total, labeled TOTAL, represents the total number of different CRF pages validated at that level over all of the days in the date range. The second, labeled FOR VALIDATIONS, is the total number of validations at that level for which it was possible to determine the time required to perform the validation. This total is used to calculate the average amount of time per validation at each validation level. All validations are counted and, as a result, each CRF may be counted more than once in this total. The third total, labeled TOTAL VALIDATIONS, is the total number of times a CRF was validated at each level (and includes those cases for which the time required to perform the validation could not be determined. This total is multiplied by the average time required

to validate records at that level to get the `EST TOTAL TIME` (estimated total time in hours) spent by the user validating records at each level.

## Validations in < 5 seconds

A separate count is also shown for the number of validations which were performed in less than 5 seconds at each validation level. Large counts in this total would tend to indicate either a very brief review of the CRFs or that the **Select > Batch Validate** feature in **DFExplore** was being abused to change the validation level of a group of records without any visual review.

## Options

<code>-u userlist</code>	Select records journaled by specified users (required)
<code>-t yy/mm/dd-yy/mm/dd-t</code>	Select records that were journaled in the specified date range, inclusive (required)
<code>-p #, #-#</code>	Select specified plate(s)
<code>-x #, #-#</code>	Select specified sequence/visit(s)
<code>-v #, #-#</code>	Select records validated to specified level(s)
<code>-f #</code>	Exclude from the statistics any records that were validated in less than # seconds
<code>-days</code>	Omit the day by day statistics from the summary statistics
<code>-status</code>	Omit the final CRF status section of the report
<code>-test</code>	Test mode. Echo the options selected without performing the report.

## Examples

### Example 2.118. Summarize CRFs validated by user1 during Jan 2016

```
-u user1 -t 160101~160131
```

### Example 2.119. Summarize CRFs validated by user1 between Dec 1,2015 and Apr 1,2016

The following strings supplied to `-t` are all equivalent ways of selecting the same records by date range.

```
-u user1 -t 151201~160401
-u user1 -t 20151201~20160401
-u user1 -t 2015/12/01~2016/04/01
```

### Example 2.120. Summarize CRFs validated by user1 during the first half of June 2015

```
-u user1 -t 2015/06/01-2015/06/15
DF_WFcrfs: Work Flow - CRFs Validated per Day.    DFstudy 65. Nov 27,2015 14:31
CRF PAGES PROCESSED PER DAY FROM 2015/06/01 to 2015/06/15 BY user1
PRIMARY RECORD VALIDATIONS PERFORMED    FINAL STATUS OF ALL CRFS
```

DATE	FROM	TO	V1	V2	V3	V4	V5	V6	V7	TOTAL	PRf	PRi	PRp	SEC	DEL	
TOTAL																
150605	09:17	15:34	109							109		109		1		
110																
150606	09:05	10:54	60							60		59		33		
92																
150608	08:41	09:23	58							58		58		4		
62																
150609	09:42	12:18	60							60		60		1		
61																
150612	09:02	09:16	9							9		7		3		
10																
150613	11:57	12:44	109							109		109		7		
116																
TOTAL			405								405		402		49	
451																
MEAN / DAY			68								68		67		8	
75																
VALIDATION TIME hrs			3.0								3.0					
FOR VALIDATIONS			398								398					
SEC / VALIDATION			27								27					
VALIDATIONS < 5sec																
TOTAL VALIDATIONS			409								409					
EST TOTAL TIME hrs			3.1								3.1					

This summary is typical for a user that validates new records to level 1 only.

## Name

DF\_WFcrfsperwk — List and graph incoming fax traffic

## Synopsis

```
DF_WFcrfsperwk {study} { [-f #] | [-f all] | [-l #] | [-l all] | [-t yyyyww~yyyyww] | [-t all] } [-g #] [-data file] [-delay #]
```

## Description

This report lists and graphs the average number of fax pages received per week.

The report counts all CRF and Query Report pages that arrive by fax during a specified time period, which is required. The time period must be specified using only one of `-f`, `-l` or `-t` option. Report output that includes all weeks of the study is specified with the keyword `all`, as in `-f all`, `-l all` or `-t all`.

The arrival week for a fax page is determined from the DFdiscover image name which is in the format `yyww` where `yy` is the 2-digit year of the century and `ww` is the 2-digit week<sup>1</sup> of the year. For example, 1701 represents the 1<sup>st</sup> week of 2017, 1801 the 1<sup>st</sup> week of 2018, and 1501 the 1<sup>st</sup> week of 2015.

The report includes all primary and secondary pages for all plates (including the returned Query Report plate), plus pages that have not yet been validated.

Only weeks during which documents were received are listed and graphed. If the number of weeks counted with new fax entries is less than the number of weeks requested, the mean number of pages per week is shown based on both the number of weeks requested *and* the number of weeks actually found.

Remember that weeks are calculated in 7 day increments from the first day of each year. As a result of this, studies which start or end in the first or last week of a year may report proportionately fewer documents received than the average number of documents per week.

To count pages received in the first few weeks of the study, use `-f`. The report considers the year and week of the first ICRed record in the journal files as the first week of the study. It is possible for this to be the wrong value, if at the start of the study, documents collect in the new image queue and the first fax to be validated is not one of the documents that arrived in the first week of the study. If this situation is suspected, use `-t` to explicitly set the date range to count pages that arrived in earlier weeks.

When `-l` is used to count pages received in the last few weeks, the current week from the computer system clock is used as the last week of the time period. However, the report only counts *complete* 7 day weeks and so the current week will typically not be included in the report listing as it is unlikely (1 in 7 chance) that the current day is also the last day of a complete week.

To write the results to a data file, in addition to the usual screen output, use `-data`. In such a case, the first line written to the specified output file identifies the column headings

```
Year | Week | Pages
```

and the data follows on subsequent lines, where `|` delimits the columns of each line. Choosing this option does not suppress the usual screen output.

## Options

**-f #-f all**                    Select the first # weeks (from the first fax received), or all weeks

<sup>1</sup> Within DFdiscover, weeks are 7 day periods beginning on the the first day of the year. The 52nd week of the year thus ends on day 364(7x52). Faxes arriving on day 365 (or 366 on leap years) have image names ending in 53. In this report pages arriving in week 53 are added to week 52, resulting in week 52 reporting 8 or 9 days of fax arrivals.



- l #-l all**                    Select the last # weeks (counting the current week as the last week), or all weeks
- t yyyyww~yyyyww-  
t all**                        Select all weeks in the specified range of weeks, inclusive, or all weeks of the study
- g #**                         In the text-only mode, set the graphics scaling to one displayed \* per # pages (default = 10).  
  
In the html mode, set the maximum value of x-axis to be # (default = the largest number of CRFs transmitted per week). To avoid decimal numbers in x-axis, it will be rounded up to the closest number which can be divided by 5 or 10. If -g takes a number less than the maximum number of CRFs transmitted per week, an arrow will appear at the end to indicate it goes beyond the x-axis maximum value.
- data file**                 Write data values used for graphing to the specified file. The filename must be supplied as an absolute pathname. The report output still appears as usual.
- delay #**                    Specify the maximum delay in validating new records (default=999999)

## Examples

### Example 2.121. Summarize the fax traffic for the last 10 weeks

```

-l 10
DF_WFcrfisperwk: CRF Pages Received Per Week.      DFstudy 65. Nov 27,2017 15:24

CRF PAGES RECEIVED PER WEEK  FOR STUDY #65  Demo Study
INCLUDING ALL CRF PAGES RECEIVED TO DATE

2017:40      253 *****
2017:41      20  **
2017:42      2
2017:46      106 *****

    Total      381 pages in 10 weeks requested but 4 weeks found
    Mean       38 pages per week requested
    Mean       95 pages per week found

Each * represents 10 pages (after rounding to the nearest 10 pages).
```

## Limitations

The documents that arrive in one week may only appear in the journal of a subsequent month. This can occur if pages sit in the unidentified router for a while before being processed. The report thus processes all of the journals starting at the beginning of the date range, looking for records that were processed by the ICR software in the time period specified. This will potentially degrade the performance of the report as it searches journal files beyond the end of the time period. The `-delay` option can be used to limit how far past the end of the date range is searched, gaining back some of the speed of the report.

## Name

**DF\_WFdiskusage** — List and graph weekly disk usage

## Synopsis

```
DF_WFdiskusage {study} [-a] [-r] [-g #] [-f week] [-l week]
```

## Description

This report lists and graphs, in chronological order, the total and average disk usage by week for study CRF images.

**DF\_WFdiskusage** lists and graphs disk usage by week. For each week, the week number, the number of Kbytes of disk used, and a histogram representation of the Kbytes used, is displayed. Weeks in which there was no disk usage are never displayed, but they can influence the calculation of the average as described below.

Summary statistics, including the total number of Kbytes, the average number of Kbytes per week, and the minimum and maximum Kbytes, are presented after the histogram. The minimum and maximum usage weeks are also indicated on the histogram by the abbreviations `min` and `max` respectively. In the case where more than one week has the minimum or maximum usage, only the first such week is indicated. Weeks in which there was no disk space used are excluded from the minimum calculation. Zero usage weeks are also excluded from the calculation of the average Kbytes used per week, unless `-a` is given. If `-a` is given, the total number of weeks considered in the average calculation is the number of weeks between the first and last week inclusive. Otherwise, only those weeks with non-zero usage are counted, which is the default.

Study weeks are specified with a `yyww` notation where `yy` is the 2-digit year of the century and `ww` is the 2-digit week of the year. By default, the range of weeks begins with the first week that a fax image was received and ends with the last week that an image was received. If `-a` is given, the last week is taken as the current week. These defaults can be overridden by specifying either or both of `-f` and `-l`.

By default, weeks are listed in chronological order, least recent first. If `-r` is specified, weeks are listed in reverse chronological order, most recent first.

Scaling of the graph can be overridden with `-g`. The value specifies the number of Kbytes represented by each `*` character. By default, the scaling factor is calculated to achieve full use of an 80-column page.

## Options

- `-a` Include all weeks in the calculation of the mean, including those weeks for which no CRFs were received
- `-r` Display weeks in reverse chronological order (most recent week first)
- `-g #` Set graphics scaling to one `*` per `#` Kbytes
- `-f week` Include only weeks on or after the specified week
- `-l week` Include only weeks before or on the specified week

## Examples

**Example 2.122.** Display, in reverse chronological order, the disk usage for study weeks from the current week back through the first week of 2017.

```
-f 0001 -r
```

DF\_WFdiskusage: Disk Space Used Per Week. DFstudy 65. Nov 27,2017 15:50

Disk usage from week 201701 to 201746 inclusive

Yr	Wk	Kb	Each * represents 300 Kb
2017	46:	2456	*****
2017	42:	43	min
2017	41:	493	*
2017	40:	5587	*****
2017	34:	1248	****
2017	33:	586	*
2017	32:	2442	*****
2017	28:	508	*
2017	27:	14136	*****max
2017	24:	2456	*****
2017	23:	5981	*****
2017	22:	1417	****
2017	21:	4645	*****
2017	20:	2355	*****
2017	19:	1529	*****
2017	17:	1196	***
2017	16:	1926	*****
2017	15:	4251	*****
2017	14:	6350	*****
2017	11:	461	*
2017	07:	165	
2017	06:	914	***
2017	03:	1488	****
2017	01:	1126	***

Total: 63759 Kbytes in 24 weeks  
 Mean: 2657 Kbytes per week  
 Min: 43 Kbytes (week 201742)  
 Max: 14136 Kbytes (week 201727)

**Example 2.123. Display, the disk usage from the 34<sup>th</sup> week of 1999 to the 34<sup>th</sup> week of 2017, inclusive**

**-f 9934 -l 0034**

DF\_WFdiskusage: Disk Space Used Per Week. DFstudy 65. Nov 27,2017 15:52

Disk usage from week 201649 to 201734 inclusive

Yr	Wk	Kb	Each * represents 300 Kb
2016	49:	2521	*****
2016	50:	2164	*****
2017	01:	1126	***
2017	03:	1488	****
2017	06:	914	***
2017	07:	165	min

```

2017 11:      461 |*
2017 14:     6350 |*****
2017 15:     4251 |*****
2017 16:     1926 |*****
2017 17:     1196 |***
2017 19:     1529 |*****
2017 20:     2355 |*****
2017 21:     4645 |*****
2017 22:     1417 |****
2017 23:     5981 |*****
2017 24:     2456 |*****
2017 27:    14136 |*****max
2017 28:      508 |*
2017 32:     2442 |*****
2017 33:      586 |*
2017 34:     1248 |****
                +-----+-----+-----+-----+-----+
Total:  59865 Kbytes in 22 weeks
Mean:   2721 Kbytes per week
Min:    165 Kbytes (week 201707)
Max:    14136 Kbytes (week 201727)

```

Note how the start of the range was automatically moved forward to the 49<sup>th</sup> week of 2016 as a result of their being no received documents before that.

## Name

**DF\_WFfaxes** — Summary of elapsed time to 1<sup>st</sup> validation

## Synopsis

```
DF_WFfaxes {study} { [-t yy/mm/dd-yy/mm/dd] | [-t ] } [-i #] [-s]
```

## Description

This report summarizes the time required to validate new records (to validation level 1) for all documents received during the specified, required time period.

All CRF pages and returned Query Reports received by fax are included. Records that subsequently attain either primary or secondary status are included independent of their status.

The information provided for each fax (or EDC data entry set) is displayed under the following columns:

FAX# <sup>①</sup>	#PAGES <sup>②</sup>	FAX ARRIVAL <sup>③</sup>	DATA ENTRY <sup>④</sup>	DELAY(hr) <sup>⑤</sup>	ENTRY(min) <sup>⑥</sup>	Min/Page <sup>⑦</sup>
-------------------	---------------------	--------------------------	-------------------------	------------------------	-------------------------	-----------------------

- ① The identification of the fax using DFdiscover notation. The identification is in the form *yywwssss* where *yy* is the 2-digit year of the century, *ww* is the week of the year, and *ssss* is the sequence number within the week of the year.
- ② The number of pages from that fax that were entered into the study database.
- ③ The date and time of fax arrival.
- ④ The date and time that the first page from the fax was validated.
- ⑤ The difference, expressed in hours, between the date and time of fax arrival, and the date and time of first page validated.
- ⑥ The total entry time, in minutes, for all of the pages in the fax <sup>1</sup>.
- ⑦ The average entry time, in minutes per page, for each page in the fax.

In addition to the per fax information, the report also displays per plate information under the heading `SUMMARY BY PLATE`. For each defined plate, a summary is displayed showing:

PLATE	PAGES <sup>①</sup>	ENTRY( N <sup>②</sup> , MEAN MIN/PAGE <sup>③</sup> )
-------	--------------------	--

- ① The total number of fax pages received.
- ② The number of these pages which contribute to the calculation of data entry time.
- ③ The average number of minutes required to validate each new record for that plate.

These statistics are useful for comparing the amount of time required to enter different CRF plates.

## Options

- t yy/mm/dd-yy/mm/dd-t** Select period when documents were received, inclusive (required)
- i #** Ignore lapse times >= #min (default=5)
- s** Create plate summary statistics only (mean data entry time by plate)

<sup>1</sup> When new records are validated in **DFexplore** they receive a record creation time stamp, which records the completion of validation, not the start. Thus, for one page documents there is only one time stamp and hence the time required to enter that page can not be calculated. For documents with more than one page the time elapsed between the creation time stamps on successive pages is summed for all pages in the fax (except the first one entered). This is used to calculate the average amount of time spent validating new records.

## Examples

### Example 2.124. Summarize processing time for documents received in the first half of November 2017

```

-t 17/11/01-17/11/15
DF_WFfaxes: 00/11/01-00/11/15 DFstudy 65. Nov 27,2017 20:10 Page 1

FAX# #PAGES FAX ARRIVAL DATA ENTRY DELAY(hr) ENTRY(min) Min/Page
17460004 11 17/11/13 11:11 17/11/14 08:51 21.7 9.7 1.0
17460005 13 17/11/13 11:21 17/11/14 09:02 21.7 7.7 0.6
17460006 1 17/11/13 11:43 17/11/14 09:08 21.4 ①
17460007 46 17/11/13 13:28 17/11/14 09:12 19.7 34.7 0.8
17460008 2 17/11/13 15:03 17/11/14 10:16 19.2 1.5 1.5
17460009 27 17/11/14 10:08 17/11/14 10:55 0.8 2.4 0.1
17460010 3 17/11/15 17:33 17/11/16 08:53 15.3 1.4 0.7

SUMMARY BY PLATE
PLATE PAGES ENTRY( N, MEAN MIN/PAGE )
1: 13 11 0.46
4: 1 1 1.00
5: 12 10 0.55
7: 6 4 0.77
8: 3 3 0.64
9: 4 4 0.27
10: 20 18 1.32
11: 7 6 0.68
12: 1 1 0.55
14: 3 2 0.86
15: 5 5 1.30
50: 1 1 0.64
501: 27 26 0.09
TOTAL: 103 92 0.62

```

① As previously noted, the entry time is calculated from the inter-page difference in creation time stamps. For this reason, single page documents cannot be calculated.

## Limitations

The report assumes that data management staff will validate one fax at a time, validating all pages in the fax before taking a break or getting another fax. It will fail to report accurate statistics if staff do not proceed in this way.

To overcome some of the inaccuracies created when the pages in a fax are not all validated at the same time, or when staff take breaks or stop to answer a phone call, the report can ignore pages which appear to have taken an unreasonable amount of time to validate. This is done by supplying `-i`. The default interval is 5 minutes. If the elapsed time from the creation time of the preceding page to the creation time of the current one is greater than this, it will not be added to the total elapsed time for that fax, and will not be included in the reported total entry time, or in the calculation of the mean number of minutes required per page. To include all entries, however long, simply specify a very large number (e.g., `-i 999999` which translates to 694 days), but remember that documents which were partially processed one day and completed days later will appear to have taken many hours to validate per page.

## Name

DF\_WFqcs — Summary of daily journal transactions for queries

## Synopsis

```
DF_WFqcs {study} [ [-t yy/mm/dd-yy/mm/dd] | [-t ] ] [-u users] [ [-q all] | [-q ext] | [-q int] ] [-P #, #-#] [-S #, #-#] [-v #, #-#] [-test]
```

## Description

This report shows the number of queries validated on each day during the specified time period for each of the requested users. Separate counts are provided for queries that are created, modified, resolved, and deleted each day. The report headings are as follows:

EXTERNAL QUERIES <sup>1</sup>					INTERNAL QUERIES <sup>2</sup>							
DATE <sup>3</sup>	FROM <sup>4</sup>	TO <sup>5</sup>	NEW <sup>6</sup>	MOD <sup>7</sup>	RES <sup>8</sup>	DEL <sup>9</sup>	TOT <sup>10</sup>	NEW	MOD	RES	DEL	TOT

- 1 Separate counts are provided for external and internal queries. This set of counts is for external queries.
- 2 This set of counts is for internal queries.
- 3 The date of the changes in *yy/mm/dd* format. Only those dates for which changes occurred are reported.
- 4 The time of the first change on the specific day. Times are reported in *hh:mm* 24-hour notation.
- 5 The time of the last change on the specific day.
- 6 The number of new queries added.
- 7 The number of previously existing queries that were modified. This may include queries that were previously created on the same day.
- 8 The number of queries resolved.
- 9 The number of queries deleted.
- 10 The total number of query changes on the day. Note that the total number of individual queries involved will be less than this total if some queries had more than one change on the day.

## Options

- t *yy/mm/dd-yy/mm/dd-t*** Include only queries journaled during the specified date range, inclusive
- u *users*** Include only queries journaled by specified users
- q *all-q ext-q int*** Select only queries having a usage from the list:  
  - all*=both internal and external (default)
  - ext*=external
  - int*=internal
- P *#, #-#*** Include only the specified plate(s)
- S *#, #-#*** Include only the specified sequence/visit(s)
- v *#, #-#*** Include only queries validated to the specified level(s)
- test** Test mode. Show options selected and exit without performing report

## Examples

**Example 2.125. Show all queries journaled by user1 from Jan 6 to Jan 9/18**

```
-u user1 -t 180106~180109
```

```
DF_WFqcs WORK FLOW FOR ALL QUERIES
```

DATE	FROM	TO	EXTERNAL QUERIES				INTERNAL QUERIES					
			NEW	MOD	RES	DEL	TOT	NEW	MOD	RES	DEL	TOT
180106	10:22	16:13	12	2	23	26	63			2		2
180107	09:10	17:26	112	1	3		116	5		30		35
180108	08:49	09:12	5		7	3	15	1				1
180109	12:00	14:17	23	2	6	9	40	2	1	3		6
TOTAL			152	5	39	38	234	8	1	35		44

This example shows that on Jan 6, 2018 user1 made a total of 63 journal entries for external queries and a total of 2 journal entries for internal queries. The first query journal entry of the day occurred at 10:22 in the morning and the last one occurred at 4:13 in the afternoon. Of the 63 journal entries for external queries, 12 represented new queries added by user1, 2 were modifications, 23 were query resolutions and 26 were query deletions. Both of the internal queries were resolved.

Note that each journal entry is counted. Thus if a query is added, modified and resolved by user1 on the same day, it is counted as 3 entries in this report.



## Name

**DF\_XXkeys** — Export keys and visit dates from required plates

## Description

**DF\_XXkeys** extracts all visit dates and key fields from required plates in the database to create summary files used by other standard report programs. Data records with status final, incomplete, missed or pending are included. **DF\_XXkeys** also checks for date inconsistencies (i.e. multiple dates for the same visit number), illegal dates, and duplicate primary database records. If any of these problems are found a warning message is displayed and a DFdiscover retrieval file is created.

**DF\_XXkeys** creates summary files in the study `WORKING_DIR` directory which are used by other standard report programs including: **DF\_QCupdate**, **DF\_QCreports**, **DF\_XXtime**, **DF\_ICvisitdates**, and **DF\_PTcrfs**. Typically, **DF\_XXkeys** is not run by itself as it creates no report output. Its value is in the summary files that it creates in preparation for other reports to run.

The summary files created by **DF\_XXkeys** are:

- `DFX_keys` (one record from each required plate that contains: ID,plate,visit,status,valid)
- (all visit dates: ID, visit, date, plate)
- (first record per visit from )
- (DRF of visit date inconsistencies)
- (DRF of illegal visit dates)
- (DRF of duplicate primary records)

When checking for illegal and inconsistent dates, **DF\_XXkeys** first determines which dates are illegal or invalid and includes them in the DRF file . Next, **DF\_XXkeys** evaluates all valid VisitDates for inconsistencies. If a VisitDate is not a valid date, it will not be reported in but will appear in only.

**DF\_ICvisitdates** reports the information found in the last 3 DRFs, namely , , and .

**DF\_XXkeys** is run automatically by **DF\_QCupdate**, **DF\_PTcrfs** and **DF\_XXtime** to update the summary files if the database or any of the relevant study configuration files (`DFschema`, `DFvisit_map`, `DFcplate_map` or `DFcterm_map`) have changed since **DF\_XXkeys** was last executed.

## Options

None.

## See Also

[DF\\_ICvisitdates](#)  
[DF\\_PTcrfs](#)  
[DF\\_QCupdate](#)  
[DF\\_QCreports](#)  
[DF\\_XXtime](#)

## Name

**DF\_XXtime** — Prepare timing summary files for other reports

## Synopsis

```
DF_XXtime {study} [-x ]
```

## Description

**DF\_XXtime** is the initialization program for other standard report programs including: **DF\_CTcrfs**, **DF\_CTqcs**, and **DF\_WFfaxes**. Typically, **DF\_XXtime** is not run by itself as it creates no report output. Its value is in the summary files that it creates in preparation for other reports to run.

**DF\_XXtime** extracts creation and modification date/time stamps from

- all primary and secondary data records from all user-defined plates (which excludes plate 501 and 511),
- the fax log, and
- new records not yet validated and calculates julian dates and times which are written to summary files in the study `WORKING_DIR` directory.

**DF\_XXtime** also requires a complete and error-free sites database to execute successfully. If **DF\_XXtime** detects errors with **DFcenters** or any other files on which it relies, it will abort and an error message will be printed to make it clear that the `DFX_dfin` summary files described below cannot be updated.

The summary files that are created by **DF\_XXtime** are:

- See [Programmer Guide, DFX\\_time1 - date and time from receipt to last modification](#) for details of the file format and contents.
- **DFX\_dfin.** This file contains one record for each fax which has been received but not yet been validated.

**DF\_XXtime** will also run **DF\_XXkeys** to rebuild if the database has been modified since the last execution of **DF\_XXkeys**. This step can be skipped by including `-x`, in which case the current version of , produced by the last execution of **DF\_XXkeys**, will be used.

## Options

None.

## Name

**DF\_qcsbyfield** — Tabulate and summarize queries by category

## Synopsis

```
DF_qcsbyfield {study} [-P # #~#] [-S # #~#] [-F # #~#] [-v # #~#] [-c # #~#] [-u i|e] [-s 0|1|2|3|4|5|6] [-t 1|2|3|4|5|6|30~99] [-D string] [-n #] [-r filename]
```

## Description

**DF\_qcsbyfield** tabulates the number of queries, of each category, for each field in the database.

The output from **DF\_qcsbyfield** is a table where each row is a unique data field and plate number combination (the rows are sorted by increasing plate number and by increasing field number within plate). The columns have the following appearance and meaning:

```
PLATE1 FIELD DESCRIPTION2 TOTAL3 MISS4 RANG5 CONS6 LEGI7 FAXN8 OTHR9 USER10
```

- 1 Plate number. Rows are sorted by increasing plate number independent of the selection order specified by `-P`.
- 2 The field number and descriptive label. Rows are sorted by increasing field number within plate number. The text of the descriptive label is the same text required by `-D string`.
- 3 The total number of queries.
- 4 The total number of queries that have a category of *missing*.
- 5 The total number of queries that have a category of *illegal*.
- 6 The total number of queries that have a category of *inconsistent*.
- 7 The total number of queries that have a category of *illegible*.
- 8 The total number of queries that have a category of *fax noise*.
- 9 The total number of queries that have a category of *other*.
- 10 The total number of queries that have any user-defined category.

Rows where the values in each tabulated column equal 0 are always omitted from the output.

## Options

- P # #~#** Select queries for specified plates
- S # #~#** Select queries for specified sequence/visit numbers
- F # #~#** Select queries for specified field numbers
- v # #~#** Select queries that are at specified validation levels
- c # #~#** Select queries for specified sites
- u i|e** Select queries by their usage category from the coded list:
  - `-u i`: internal queries
  - `-u e`: external queries
- s 0|1|2|3|4|5|6** Select queries by one status code from the list: 0=pending review, 1=new, 2=in unsent report, 3=resolved, NA, 4=resolved, irrelevant, 5=resolved, corrected, 6=in sent report

- t 1|2|3|4|5|6|30~99** Select queries by one category from the list: 1=missing, 2=illegal, 3=inconsistent, 4=illegible, 5=fax noise, 6=other, 30-99=user-defined category
- D string** Select queries containing *string* in the variable description. The description *string* must exactly match (with case sensitivity) the entire string (or a sub-string) of the query description label. If the match string contains embedded spaces, it must be enclosed in ", e.g. -D "Prior Rx".
- n #** Select only data fields which have # or more queries
- r filename** Write each selected case to the named DRF. If *filename* does not start with /, it is created relative to the study *drf* directory.

## Examples

### Example 2.126. Tabulate external queries on plates 3 through 7 and 10

```
-P 3~7 10 -u e
DF_qcsbyfield: Queries on Selected Data Fields.  DFstudy 253.  Jul 06,2016 10:23
```

PLATE	FIELD DESCRIPTION	TOTAL	MISS	RANG	CONS	LEGI	FAXN	OTHR	USER
3	8. Patient Initials	3:	0	0	0	0	0	2	1
4	8. Patient Initials	2:	0	0	0	0	0	2	0
5	8. Patient Initials	11:	0	0	0	0	0	9	2
5	10. Reading 1 systolic BP	1:	1	0	0	0	0	0	0
5	11. Reading 1 diastolic BP	1:	1	0	0	0	0	0	0
5	12. Reading 2 systolic BP	1:	1	0	0	0	0	0	0
5	13. Reading 2 diastolic BP	1:	1	0	0	0	0	0	0
5	16. Weight in lbs	5:	1	0	0	0	0	0	4
5	19. Returned Med. Code #	1:	1	0	0	0	0	0	0
5	23. New Bottle Med. Code #	2:	1	0	0	0	0	1	0
7	8. Patient Initials	5:	0	0	0	0	0	3	2
7	10. 1. complete entire study	1:	1	0	0	0	0	0	0
10	8. Patient Initials	1:	0	0	0	0	0	1	0

### Example 2.127. Tabulate fields which have 5 or more queries

```
-n 5
DF_qcsbyfield: Queries on Selected Data Fields.  DFstudy 253.  Dec 04,2015 09:12
```

PLATE	FIELD DESCRIPTION	TOTAL	MISS	RANG	CONS	LEGI	FAXN	OTHR	USER
2	8. Patient Initials	5:	0	0	0	0	0	5	0
2	11. Date of Birth	5:	4	0	1	0	0	0	0
2	19. Date of First Follow-up	6:	6	0	0	0	0	0	0
5	8. Patient Initials	11:	0	0	0	0	0	9	2

### Example 2.128. Tabulate queries for fields with "BP" in their description

```
-D BP
```

DF\_qcsbyfield: Queries on Selected Data Fields. DFstudy 253. Aug 11,2016 14:25

PLATE	FIELD	DESCRIPTION	TOTAL	MISS	RANG	CONS	LEGI	FAXN	OTHR	USER
1	29.	3. Taking BP medication	2:	2	0	0	0	0	0	0
5	10.	Reading 1 systolic BP	1:	1	0	0	0	0	0	0
5	11.	Reading 1 diastolic BP	1:	1	0	0	0	0	0	0
5	12.	Reading 2 systolic BP	1:	1	0	0	0	0	0	0
5	13.	Reading 2 diastolic BP	2:	1	0	0	0	0	0	1

## See Also

[DF\\_SSschema](#)

## Name

**DF\_stats** — Display simple variable statistics for a single plate

## Synopsis

```
DF_stats {study} {-P #} [-n #, #-#] [-S #, #-#] [[-F #, #-#] | [-N #, #-#]] [-G visit plate field code1 label1
code2 label2] [-z] [-L lines] [-in file] [-ex file]
```

## Description

**DF\_stats** displays simple descriptive statistics for fields on a specified plate.

**DF\_stats** displays descriptive statistics for fields on a specified plate. It includes all final, incomplete and pending records for the specified plate; it does not include secondary records.

The descriptive statistics displayed depend on the field type and include:

- **Choice.** number of cases for each choice code
- **Check.** number of cases checked and not checked
- **Numeric (including VAS).** mean, standard deviation, and range
- **Date.** date range <sup>1</sup>
- **Text.** minimum and maximum length

For all fields, missing data counts are displayed:

- **Missing Values.** fields marked with a missing value code
- **Illegal Values.** fields that fail the legal range specification from the study schema
- **Blank Values.** fields that contain nothing. This does not include choice or check fields where the response is no choice or not checked
- **Missing Records.** cases (in a group specification) for which data is not available

For fields of date type, **DF\_stats** also tabulates the number of *invalid* values, where invalid means that one or more the day, month, or year parts is impossible or nonsensical.

The specification of field numbers uses study schema field numbers which include the DFdiscover-defined fields (*DFstatus*, *DFvalid*, etc.) in addition to user-defined data fields. Use **DF\_SSschema** or **DF\_SSvars** to display field numbers, variable names, description, coding, etc.

When using **-G**, the code and label to be used for each group are required. It is possible to specify two groups only. Only subjects meeting the grouping specifications will be included in the report. The number of subjects categorized into both groups must be greater than 0, or the report will fail.

### Example 2.129. Group by the code in field 9 of plate 1, visit 0

```
-G 0 1 9 1 male 2 female
```

<sup>1</sup>The date range displayed for legal dates always uses imputed values where imputation is defined.

This specification will result in 2 groups based on the grouping variable found in field 9 of plate 1. Cases with the value 1 will go into the male group and cases with code 2 will go into the female group. Any cases that have neither of these codes will not be included.

It is legal to specify a group comprised of more than 1 code. The codes that comprise each group are specified as a list of values and ranges, using the standard format #, #-#.

### Example 2.130. Group by range of ages

```
-G 0 1 9 0~15 child 16~99 adult
```

This example creates 2 groups on the basis of age (which can be found on visit 0, plate 1, field 9). Subjects 15 and under are grouped as children while those 16 and older are grouped as adults. Notice that there is no syntax for specifying an open-ended range. In this example, the upper-limit of 99 on adults would exclude subjects whose age is 100 or greater from the reported results.

If `-G` is used to display statistics by a grouping variable, it should be used in conjunction with `-S`. The percentages shown will then be interpretable as the percentage of subjects in each group. Otherwise if more than 1 visit/sequence number is tabulated some subjects may have more records in the plate than other subjects and the percentages will be based on the total number of records rather than the total number of subjects in each group.

It is possible, for purposes of tabulation, to coerce specified string fields to be numeric fields with `-N`. This allows **DF\_stats** to report the mean, standard deviation, etc. Note however that string values which cannot be coerced to numeric will be counted as legal values but will not be included in the calculation of mean, standard deviation, or range.

Specific subjects can be included or excluded from the statistics by including their ids in an inclusion or exclusion file. In both cases, the file format is simply one id number per line. It is valid to use an inclusion file and an exclusion file and specify them both as options as in:

```
-in inclusionfile -ex exclusionfile
```

In such a case, the included subject IDs are those from the inclusion file that do not appear in the exclusion file.

## Options

<b>-P #</b>	Select the plate number (required)
<b>-n #, #-#</b>	Select only subjects from specified site IDs
<b>-S #, #-#</b>	Select only specified sequence (or visit) numbers
<b>-F #, #-#-N #, #-#</b>	Select only specified field numbers. The specified fields are coerced to numbers if <code>-N</code> is used.
<b>-G visit plate field code1 label1 code2 label2</b>	Report stats by group. The <i>visit plate field</i> values select the grouping variable and <i>code1 label1 code2 label2</i> specify the two required codes.
<b>-L lines</b>	Maximum number of lines per report page
<b>-z</b>	Exclude any coded categories that have zero counts
<b>-in file</b>	Include subject only if ID is listed in <i>file</i>
<b>-ex file</b>	Exclude subject if ID is listed in <i>file</i>

## Examples

### Example 2.131. Stats for all fields on plate 7 (for all visits)

```
-P 7
```

### Example 2.132. Stats for all fields on plate 7 at visit 0

```
-P 7 -S 0
```

### Example 2.133. Stats for fields 8 through 12 inclusive on plate 22 (for all visits)

```
-P 22 -F 8~12
```

### Example 2.134. Group plate 6 observations by sex (which is visit 0 plate 1 field 9)

```
-P 6 -G 0 1 9 1 male 2 female
```

## Limitations

**DF\_stats** is limited to a maximum of 100 fields at a time. On plates which have more than 100 fields use `-F` iteratively to specify a set of 100 fields or less for each execution.

## See Also

[DF\\_SSschema](#)

[DF\\_SSvars](#)



# Appendix A. Copyrights - Acknowledgments

## A.1. External Software Copyrights

DFdiscover software uses several third-party software components as part of its server side and/or client tools.

The copyright information for each is provided below. If you would like to receive source codes of these third-party components, please send us your request at <help@dfnetresearch.com>.

### A.1.1. DCMTK software package

Copyright Copyright © 1994-2011, OFFIS e.V. All rights reserved.

This software and supporting documentation were developed by

OFFIS e.V.

R&D Division Health  
Eschereg 2  
26121 Oldenburg, Germany

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of OFFIS nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

### A.1.2. Jansson License

Copyright Copyright © 2009-2014 Petri Lehtinen <petri@digip.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,

DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### **A.1.3. Mimencode**

Copyright Copyright © 1991 Bell Communications Research, Inc. (Bellcore)

Permission to use, copy, modify, and distribute this material for any purpose and without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Bellcore not be used in advertising or publicity pertaining to this material without the specific, prior written permission of an authorized representative of Bellcore. BELLCORE MAKES NO REPRESENTATIONS ABOUT THE ACCURACY OR SUITABILITY OF THIS MATERIAL FOR ANY PURPOSE. IT IS PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES.

### **A.1.4. RSA Data Security, Inc., MD5 message-digest algorithm**

Copyright Copyright © 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function. License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work. RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind. These notices must be retained in any copies of any part of this documentation and/or software.

### **A.1.5. mpack/munpack**

Copyright © Copyright 1993,1994 by Carnegie Mellon University All Rights Reserved.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Carnegie Mellon University not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Carnegie Mellon University makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

### **A.1.6. TIFF**

Copyright Copyright © 1988-1997 Sam Leffler Copyright Copyright © 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL,

INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## A.1.7. PostgreSQL

Portions Copyright © 1996-2019, PostgreSQL Global Development Group Portions Copyright © 1994, The Regents of the University of California

Permission to use, copy, modify, and distribute this software and its documentation for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

## A.1.8. OpenSSL License

Copyright Copyright © 1998-2019 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in .the OpenSSL Toolkit." (<http://www.openssl.org/>)
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [openssl-core@openssl.org](mailto:openssl-core@openssl.org).
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (<eay@cryptsoft.com>). This product includes software written by Tim Hudson (<tjh@cryptsoft.com>).

## A.1.9. Original SSLeay License

Copyright Copyright © 1995-1998 Eric Young (<eay@cryptsoft.com>) All rights reserved.

This package is an SSL implementation written by Eric Young (<eay@cryptsoft.com>). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (<tjh@cryptsoft.com>).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (<eay@cryptsoft.com>)" The word "cryptographic" can be left out if the rouines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (<tjh@cryptsoft.com>)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

## A.1.10. gawk

GNU GENERAL PUBLIC LICENSE Version 2, June 1991

<http://www.gnu.org/licenses/gpl-2.0.html>

Copyright Copyright © 1989, 1991 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor,  
Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

#### TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

1. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

2. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

3. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:



- a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.) These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

4. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
  - a. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
  - b. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
  - c. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.) The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

5. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
6. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your

acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

7. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
8. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

9. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
10. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

11. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally. NO WARRANTY

12. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

13. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED

ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## A.1.11. Ghostscript

The files in the base, psi, lib, toolbox, examples, doc and man directories (folders) and any subdirectories (sub-folders) thereof are part of GPL Ghostscript.

The files in the Resource directory and any subdirectories thereof are also part of GPL Ghostscript, with the explicit exception of the files in the CMap subdirectory (except "Identity-UTF16-H", which is part of GPL Ghostscript). The CMap files are copyright Adobe Systems Incorporated and covered by a separate, GPL compatible license.

The files under the jpegxr directory and any subdirectories thereof are distributed under a no cost, open source license granted by the ITU/ISO/IEC but it is not GPL compatible - see jpegxr/COPYRIGHT.txt for details.

GPL Ghostscript is free software; you can redistribute it and/or modify it under the terms the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

GPL Ghostscript is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program so you can know your rights and responsibilities. It should be in a file named doc/COPYING. If not, write to the

Free Software Foundation, Inc., 59 Temple Place Suite 330, Boston, MA  
02111-1307, USA.

GPL Ghostscript contains an implementation of techniques covered by US Patents 5,055,942 and 5,917,614, and corresponding international patents. These patents are licensed for use with GPL Ghostscript under the following grant:

Whereas, Raph Levien (hereinafter "Inventor") has obtained patent protection for related technology (hereinafter "Patented Technology"), Inventor wishes to aid the the GNU free software project in achieving its goals, and Inventor also wishes to increase public awareness of Patented Technology, Inventor hereby grants a fully paid up, nonexclusive, royalty free license to practice the patents listed below ("the Patents") if and only if practiced in conjunction with software distributed under the terms of any version of the GNU General Public License as published by the

Free Software Foundation, 59 Temple Place, Suite  
330, Boston, MA 02111.

Inventor reserves all other rights, including without limitation, licensing for software not distributed under the GNU General Public License.

5055942 Photographic image reproduction device using digital halftoning to para images allowing adjustable coarseness  
5917614 Method and apparatus for error diffusion paraing of images with improved smoothness in highlight and shadow regions

## A.1.12. MariaDB and FreeTDS

GNU LESSER GENERAL PUBLIC LICENSE Version 2.1, February 1999 <http://www.gnu.org/licenses/lgpl-2.1.html>



Copyright Copyright © 1991, 1999

Free Software Foundation, Inc.  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method:

1. we copyright the library, and
2. we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

#### TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

1. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

2. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

3. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a. The modified work must itself be a software library.
- b. You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c. You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d. If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful. (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

4. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

5. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

6. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a “work that uses the Library” uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

7. As an exception to the Sections above, you may also combine or link a “work that uses the Library” with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer’s own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License.

Also, you must do one of these things:

- a. Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable “work that uses the Library”, as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
- b. Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that
  - i. uses at run time a copy of the library already present on the user’s computer system, rather than copying library functions into the executable, and
  - ii. will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.
- c. Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
- d. If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
- e. Verify that the user has already received a copy of these materials or that you have already sent this user a copy. For an executable, the required form of the “work that uses the Library” must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

- f. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
  - a. Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
  - b. Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
- g. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
- h. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.
- i. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.
- j. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

- k. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
- l. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

- m. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

- n. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

- o. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## A.1.13. QtAV

Copyright © Wang Bin <wbsecg1@gmail.com> Shanghai University->S3 Graphics->Deepin, Shanghai, China 2013-01-21

\*\*QtAV is free software licensed under the term of LGPL v2.1. The player example is licensed under GPL v3. If you use QtAV or its constituent libraries, you must adhere to the terms of the license in question.\*\*

Rather than repeating the text of the LGPL v2.1, the original text can be found in [GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1 \[159\]](#).

## A.1.14. FFmpeg

Most files in FFmpeg are under the GNU Lesser General Public License version 2.1 or later (LGPL v2.1+). Read the file `COPYING.LGPLv2.1` for details. Some other files have MIT/X11/BSD-style licenses. In combination the LGPL v2.1+ applies to FFmpeg.

Rather than repeating the text of the LGPL v2.1, the original text can be found in [GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1 \[159\]](#).

## A.1.15. c3.js

The MIT License (MIT) Copyright © 2013 Masayuki Tanaka

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify,



merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## **A.1.16. d3.js**

Copyright 2010-2017 Mike Bostock All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: \* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. \* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. \* Neither the name of the author nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Index

## C

---

c3.js, [165](#)

## D

---

d3.js, [166](#)  
DCMTK, [152](#)  
DF\_ATcrfs, [24-25](#)  
DF\_ATfaxes, [26-33](#)  
DF\_ATmods, [29](#)  
DF\_CTcrfs, [34-35](#)  
DF\_CTpages, [36](#)  
DF\_CTqcs, [37-39](#)  
DF\_CTvisits, [40-41](#)  
DF\_DatabaseDefinition, [4](#)  
DF\_DataEntryTimeTrend, [5](#)  
DF\_Enrollment, [6](#)  
DF\_ICcenters, [42-43](#)  
DF\_ICimages, [44](#)  
DF\_ICkeys, [45-48](#)  
DF\_ICqcs, [49](#)  
DF\_ICrecords, [50-51](#)  
DF\_ICschema, [52-53](#)  
DF\_ICvisitdates, [54](#)  
DF\_ICvisitmap, [55-58](#)  
DF\_InboundTraffic, [7](#)  
DF\_PlateStatus, [8](#)  
DF\_PTcrfs, [59-60](#)  
DF\_PTlist, [61-65](#)  
DF\_PTmissing, [66-67](#)  
DF\_PTqcs, [68-69](#)  
DF\_PTschedule, [70-71](#)  
DF\_PTunexpected, [72-73](#)  
DF\_PTvisits, [74-78](#)  
DF\_QCfax, [79-83](#)  
DF\_QCfaxlog, [84-85](#)  
DF\_QCprint, [86](#)  
DF\_QCreports, [88-105](#)  
DF\_qcsbyfield, [146-148](#)  
DF\_QCsent, [106-108](#)  
DF\_QCstatus, [109](#)  
DF\_QCupdate, [111-116](#)  
DF\_QCview, [117](#)  
DF\_queriesbyfield, [23](#)  
DF\_QueryReport, [9](#)  
DF\_QueryStatusTrend, [10](#)  
DF\_QueryTimeTrend, [11](#)  
DF\_QueryUpdate, [12](#)  
DF\_SBaliaslog, [17](#)  
DF\_SBaliasmap, [16](#)  
DF\_SBhistory, [13](#)

DF\_SBSchedule, [14](#)  
DF\_SBunexpected, [15](#)  
DF\_SScenters, [118](#)  
DF\_SSschema, [119-121](#)  
DF\_SSsites, [18](#)  
DF\_SSvars, [122-125](#)  
DF\_SSvisitmap, [126-131](#)  
DF\_stats, [149-151](#)  
DF\_Status, [22](#)  
DF\_STcrfs, [19](#)  
DF\_STqueries, [20-20](#)  
DF\_STvisits, [21](#)  
DF\_WFcrfs, [132-134](#)  
DF\_WFcrfsperwk, [135-136](#)  
DF\_WFdiskusage, [137-139](#)  
DF\_WFfaxes, [140-141](#)  
DF\_WFqcs, [142-143](#)  
DF\_XXkeys, [144](#)  
DF\_XXtime, [145](#)

## E

---

External Software Copyrights, [152-166](#)  
c3.js, [165](#)  
d3.js, [166](#)  
DCMTK, [152](#)  
FFmpeg, [165-166](#)  
gawk, [155-159](#)  
Ghostscript, [159-159](#)  
Jansson License, [152-153](#)  
MariaDB and FreeTDS, [159-165](#)  
Mimencode, [153](#)  
mpack/munpack, [153](#)  
OpenSSL License, [154-155](#)  
PostgreSQL, [154](#)  
QtAV, [165](#)  
RSA Data Security, Inc., MD5 message-digest algorithm, [153](#)  
TIFF, [153](#)

## F

---

FFmpeg, [165](#)  
FreeTDS, [159](#)

## G

---

gawk, [155](#)  
Ghostscript, [159](#)

## J

---

Jansson, [152](#)

## M

---

MariaDB, [159](#)



Mimencode, [153](#)  
mpack/munpack, [153](#)

## O

---

OpenSSL, [154](#)

## P

---

PostgreSQL, [154](#)

## Q

---

QtAV, [165](#)

## R

---

RSA Data Security, [153](#)

## T

---

TIFF, [153](#)