

IBM FileNet Image Services
Version 4.2

Index and WorkFlo Database Contents



IBM FileNet Image Services
Version 4.2

Index and WorkFlo Database Contents



Note

Before using this information and the product it supports, read the information in "Notices" on page 109.

This edition applies to version 4.2 of IBM FileNet Image Services (product number 5724-R95) and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 1984, 2011.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this manual 9

| | |
|-------------------------------------|----|
| Document revision history | 10 |
| Manual content | 10 |
| Relational database support | 11 |
| IBM DB2 | 11 |
| Oracle | 12 |
| MS SQL Server | 12 |
| Accessing IBM FileNet Documentation | 12 |
| IBM FileNet Education | 13 |
| Feedback | 13 |
| Documentation feedback | 13 |
| Product consumability feedback | 13 |

1 Introduction 14

2 Working with Oracle 16

| | |
|--------------------------------|----|
| Using SQL*Plus | 17 |
| Logging on | 17 |
| Changing the password | 18 |
| Listing tables | 18 |
| Listing the columns of a table | 20 |

Viewing user index names 23

Managing WorkFlo queues 24

Mapping queue column names to field names 25

Saving WQS_tool commands and output to a file 26

Using spacerpt to view tables 27

Creating views 27

Creating WorkFlo Queue tables and views 30

Determining maximum string lengths 31

Working with dates 32

Translating an integer date to a date 32

Changing the Date Format 33

Selecting documents based on dates 34

3 Working with Microsoft SQL Server 35

Using isql 35

Changing the password 35

Listing tables 36

Listing the columns of a table 38

Viewing user index names 41

Managing WorkFlo queues 43

Mapping queue column names to field names 44

Saving WQS_tool commands and output to a file 45

Using spacerpt to view tables 45

Creating views 46

Creating WorkFlo queue tables and views 48

Determining maximum string lengths 49

4 Working with DB2 51

Using DB2 51

Logon to the database instance 52

Listing tables 52

Listing the columns of a table 56

Viewing user index names 60

Managing WorkFlo queues 62

Mapping queue column names to field names 63

Saving WQS_tool commands and output to a file 64

Creating views 65

Creating WorkFlo queue tables and views 67

Determining maximum string lengths 67

Working with dates 68

Translating an integer date to a date 68

5 Table descriptions 70

Standard tables 71

ce_id_map table 71

ce_os_dcl_map table 72

doctaba table 73

document_class table 75

doc_class_index table 80

export_log table 83

folder table 84

| | |
|-----------------------------|------------|
| folder_contents table | 86 |
| GUIDS table | 87 |
| index_cluster table | 88 |
| menu table | 89 |
| menu_items table | 90 |
| no_cat_audit table | 91 |
| sys_numbers table | 92 |
| user_index table | 93 |
| validation_tab table | 96 |
| validation_tab_items table | 96 |
| WQS Tables | 97 |
| wqs_idseed table | 97 |
| wqs_workspaces table | 98 |
| wqs_queues table | 99 |
| wqs_fields table | 100 |
| wqs_release table | 101 |
| WorkFlo Queue Tables | 102 |

Glossary 104

Notices 109

| | |
|--------------------------------|------------|
| Trademarks | 113 |
| U.S. Patents Disclosure | 113 |

Index 114

About this manual

This *Index and WorkFlo Database Contents* for IBM® FileNet® Image Services manual provides system administrators and other interested users with a description of the structure and contents of the FileNet index and WorkFlo databases.

Structured Query Language (SQL) is a standardized database query language. You must be familiar with SQL to use certain Oracle server manager functions. For more information there are many books available on SQL.

Transact-SQL is Microsoft's enhanced version of SQL for Microsoft® SQL Server™. Microsoft SQL Server is a relational database management system (RDBMS). If you are not familiar with Transact-SQL and the related tools for Microsoft SQL Server, see Microsoft's *Transact-SQL Reference* and *Microsoft SQL Server Administrator's Companion*.

We assume you are familiar with FileNet system operations and terminology as described in the following documentation:

- *IDM Desktop User's Help*
- *Image Services System Administrator's Handbook*

Document revision history

| Image Services version | Date | Comment |
|------------------------|----------|------------------|
| 4.2 | May 2011 | Initial release. |

Manual content

This manual contains the information you need to work with the data stored by FileNet® applications in an Oracle®, Microsoft® SQL Server, or IBM® DB2® RDBMS. The manual organization is described below.

Chapter 1, “Introduction,” on page 14 provides an overview of the FileNet index and WorkFlo databases, discusses the access permitted by each type of RDBMS license, and describes restrictions for accessing the FileNet database tables.

Chapter 2, “Working with Oracle,” on page 16 explains how to list Oracle database information using SQL and WQS_tool. This chapter also explains how to use the dclview tool to create a more user-friendly view of document index values, how to find the maximum length of string fields, and how to work with dates.

Chapter 3, “Working with Microsoft SQL Server,” on page 35 explains how to list Microsoft SQL Server database information using WQS_tool. This chapter also explains how to use the dclview tool to create a more user-friendly view of document index values and how to find the maximum length of string fields.

Chapter 4, “Working with DB2,” on page 51 explains how to access DB2 database information and work with tables. This chapter also explains how to use the dclview tool to create a more user-friendly view

of document index values, how to find the maximum length of string fields, and how to work with dates.

Chapter 5, “Table descriptions,” on page 70 contains detailed information on each FileNet-defined table in an index database.

Relational database support

Image Services supports IBM DB2, Oracle, and MSSQL relational databases. For information on the specific versions of the relational database software supported in this Image Services release, see the *IBM FileNet Image Services, IBM FileNet Image Services Resource Adapter, and IBM FileNet Print Hardware and Software Requirements* document. To download IBM FileNet documentation from the IBM support page, see **“Accessing IBM FileNet Documentation” on page 12**

IBM DB2

All DB2 databases are site-controlled and can be on any platform that FileNet Image Services currently supports. Image Services on the Solaris server accesses the remote DB2 database by using DB2 client software installed on the IS server.

See *Guidelines for Installing and Configuring DB2 Software* for more information. To download IBM FileNet documentation from the IBM support page, see **“Accessing IBM FileNet Documentation” on page 12**

Oracle

Oracle relational databases can be installed locally on the IS Root/ Index server, on an IS Application server, or on a remote database server. All new installations of Oracle are site-controlled.

See *Guidelines for Installing and Updating Site-Controlled Oracle Software on UNIX Servers* for more information. To download IBM FileNet documentation from the IBM support page, see [**“Accessing IBM FileNet Documentation” on page 12.**](#)

MS SQL Server

Image Services software supports Microsoft SQL Server. MS SQL relational databases are supported only on Windows servers. Database tables are read-only accessible through PC WorkFlo or Microsoft SQL Server tools.

See *Guidelines for Installing and Updating RDBMS Software on Windows Servers* for more information. To download IBM FileNet documentation from the IBM support page, see [**“Accessing IBM FileNet Documentation” on page 12.**](#)

Accessing IBM FileNet Documentation

To access documentation for IBM FileNet Image Services products:

- 1 On the [**www.ibm.com**](http://www.ibm.com) website, enter “Image Services Documentation” in the search box on the menu bar.
- 2 Select **IBM - Product Documentation for FileNet Image Services** from the list of search results.

IBM FileNet Education

IBM provides various forms of education. Please visit the IBM Training and Certification for IBM software page at:

www.ibm.com/software/sw-training.

Feedback

We value your opinion, experience, and use of our products. Please help us improve our products by providing feedback or by completing a consumability survey.

Documentation feedback

Send comments on this publication or other IBM FileNet Image Services documentation by e-mail to comments@us.ibm.com. Be sure to include the name of the product, the version number of the product, and the name and part number of the book (if applicable). If you are commenting on specific text, include the location of the text (for example, a help topic title, a chapter and section title, a table number, or a page number).

Product consumability feedback

Help us identify product enhancements by taking a [Consumability Survey](#). The results of this comprehensive survey are used by product development teams when planning future releases. Although we are especially interested in survey responses regarding the most recent product releases, we welcome your feedback on any of our products.

The survey takes approximately 30 minutes to complete and must be completed in a single session; there is no option to save a partially completed response.

Introduction

The Image Services software stores much of your data in an RDBMS database called the index database. The index database stores:

- Document index values, index field definitions, and document class definitions
- An audit trail showing each time index cataloging is turned on or off for any document class
- Menu and validation tables
- Folder definitions, and folder contents
- WorkFlo queues

On a WorkFlo Queue server, an RDBMS database stores additional WorkFlo queues.

An RDBMS database consists of a set of tables, some of which are interrelated, and a set of rules for making insertions, updates, and deletions. Each table stores data related to a set of similar entities, and the columns of a table describe the attributes of the entity.

Tip The database table ID used by index and WorkFlo Queue services is F_SW (Oracle) or f_sw (Microsoft SQL Server). The database table ID used by eProcess is F_SQL (Oracle) or f_sqi (Microsoft SQL Server). These IDs, used by the FileNet software to log on to the RDBMS, are the owners of the tables.

You can access an RDBMS database using Oracle or Microsoft SQL Server. On UNIX platforms, you can use Oracle and on Windows Server you can use either Oracle or Microsoft SQL Server.

Important Your access to the FileNet database tables is read-only to preserve the integrity of the database. The FileNet database tables must not be altered using tools other than FileNet software. Any modification by non-FileNet software could violate your support agreement and result in time and materials charges for repair of the database.

Tip An existing customer database can be incorporated into a FileNet database and a FileNet database can be incorporated into an existing customer database.

2

Working with Oracle

This chapter explains the following topics:

- How to use SQL select and describe statements and WQS_tool to find the index table and column names for use in queries (we assume that you know SQL)
- How to save WQS_tool output to a file
- How to use dclview to create views, which give you a more user-friendly view of document index values
- How to use WorkFlo queue views, which are created for you automatically
- How to work with dates and times stored in the database as numbers
- How to find the size of a field

Using SQL*Plus

If you have an Oracle license, you can use SQL*Plus to send SQL select and describe statements to Oracle.

Logging on

To log onto SQL*Plus:

1 Log on to the appropriate server.

2 Enter:

sqlplus

3 Enter the username:

f_sqi

The f_sqi username has read-only privileges.

4 Enter the password:

f_sqi

You see this prompt:

SQL>

You can now enter SQL*Plus commands.

Changing the password

To change the f_sqi password:

- 1 Log onto SQL*Plus.
- 2 Enter this command, substituting the new password for <password>:

```
alter user f_sqi identified by <password>;
```

Listing tables

A set of standard FileNet tables always exists in the index database. Table names beginning with WQM are WorkFlo queues. See [Chapter 5, “Table descriptions,” on page 70](#). To list the names of all FileNet-created tables in an index database, enter this SQL command on the Index server:

```
select owner, table_name from all_tables where owner='F_SW'  
or owner='F_SQL';
```

This statement produces a list similar to the following (on a WorkFlo Queue server, this statement lists only WorkFlo queues):

```
SQL>select owner, table_name from all_tables where
owner='F_SW' or owner='F_SQI';
```

| OWNER | TABLE_NAME |
|-------------------|----------------------|
| ----- | ----- |
| F_SW | SYS_NUMBERS |
| F_SW | USER_INDEX |
| F_SW | INDEX_CLUSTER |
| F_SW | DOCUMENT_CLASS |
| F_SW | DOC_CLASS_INDEX |
| F_SW | DOCTABA |
| F_SW | FOLDER |
| F_SW | FOLDER_CONTENTS |
| F_SW | FOLDER_TABS |
| F_SW | NO_CAT_AUDIT |
| F_SW | MENU |
| OWNER | TABLE_NAME |
| ----- | ----- |
| F_SW | MENU_ITEMS |
| F_SW | VALIDATION_TAB |
| F_SW | VALIDATION_TAB_ITEMS |
| F_SW | WQS_IDSEED |
| F_SW | WQS_WORKSPACES |
| F_SW | WQS_QUEUES |
| F_SW | WQS_FIELDS |
| 18 rows selected. | |

All tables are owned by F_SW. To list the tables in the index database, enter this SQL command:

```
select owner, table_name from all_tables where owner='F_
SW';
```

Listing the columns of a table

Users logged in as f_sqi can list the FileNet tables explicitly. Users logged in as f_maint can use sys.dba_tab_columns.

To list a table's columns, enter an SQL select statement.

```
select column_name from all_tab_columns where  
table_name='<table>';
```

Replace <table> with the table name using uppercase letters.

When you use a select statement to list the columns of the doctaba table, you see something like this:

```

SQL>select column_name from all_tab_columns where table_name='DOCTABA';

COLUMN_NAME
-----
A32
F_DOCNUMBER
F_DOCCLASSNUMBER
F_ENTRYDATE
F_LASTACCESS
F_ANNOTATIONFLAG
F_ARCHIVEDATE
F_PURGEDATE
F_DELETEDATE
F_RETENTBASE
F_RETENTDISP

COLUMN_NAME
-----
F_RETENTOFFSET
F_PAGES
F_SECURITYSPEC
F_ACCESSRIGHTS
F_DOCTYPE
F_STATUS
A31
18 rows selected.

```

If you have an Oracle license, you can list the columns of a table with an SQL describe statement in this format:

describe <table name>

Replace <table name> with the name of the desired table. For example, to list information for doctaba, enter this statement:

describe DOCTABA

This describe statement lists the name of each column in the table, whether the column requires a value (not null), and the column type.

For example:

```
SQL>describe DOCTABA
Name                                Null?                                Type
-----
F_DOCNUMBER                        NOT NULL                            NUMBER (10)
F_DOCCLASSNUMBER                  NOT NULL                            NUMBER (5)
F_ENTRYDATE                       NOT NULL                            NUMBER (10)
F_LASTACCESS                      NUMBER (10)
F_ANNOTATIONFLAG                  VARCHAR2 (1)
F_ARCHIVEDATE                     NUMBER (10)
F_PURGEDATE                       NUMBER (10)
F_DELETEDATE                      NUMBER (10)
F_RETENTBASE                      VARCHAR2 (1)
F_RETENTDISP                      VARCHAR2 (1)
F_RETENTOFFSET                   NUMBER (5)
F_PAGES                           NUMBER (5)
F_SECURITYSPEC                   VARCHAR2 (12)
F_ACCESSRIGHTS                   VARCHAR2 (12)
F_DOCTYPE                        VARCHAR2 (1)
F_STATUS                          NUMBER (5)
A31                              VARCHAR2 (239)
A32                              NUMBER
A33                              VARCHAR2 (239)
A34                              VARCHAR2 (14)
```

Viewing user index names

In doctaba, the user-defined indexes are named A31, A32, and so on. To view numbered column names as the actual index names, enter:

```
select f_columnname, f_indexname from user_index;
```

This statement produces a list similar to the one shown below. It maps the doctaba column names A31, A32, and so on, to the user-defined index names stored in the user_index table.

```
SQL>select f_columnname, f_indexname from user_index;

F_COL F_INDEXNAME
-----
      F_DOCNUMBER
      F_DOCCLASSNUMBER
      F_ARCHIVEDATE
      F_PURGEDATE
      F_DELETEDATE
      F_ENTRYDATE
      F_LASTACCESS
      F_RETENTOFFSET
      F_PAGES
      F_DOCTYPE
      F_RETENTBASE

F_COL F_INDEXNAME
-----
      F_RETENTDISP
      F_ACCESSRIGHTS
a31   test1
a32   test2

15 rows selected.
```

Managing WorkFlo queues

A WQS_tool table report gives you the names of the workspaces and queues on the server, including the server ID and table ID that relate to the queue name. The name format is **WQMsssQnnnnnn**, where **sss** is the server ID and **nnnnnn** is the table ID. For example, the table name of the Charlotte queue in the report below is **WQM001Q001011**.

For queues created under IMS version 3.0.3 or earlier, the name format is **WQM1nnnnnn**, where **nnnnnn** is the table ID listed in the report. A queue's name does not change. For example, the ApplDistQ in the report below was created under version 3.0.3, so its table name is **WQM101001**.

To enter the WQS_tool utility for managing WorkFlo queues, enter the following command at the operating system prompt:

WQS_tool

To list the workspaces, queue names, and table IDs, enter:

table * *

This displays a report:

```
<WQS_tool>table * *
Workspace      Queue name      table name      Queue Server
-----
pcwfl          ApplDistQ       01001           WflServer
SomeBank       Fax_In          01007           WflServer
SomeBank       Greensboro      01010           WflServer
SomeBank       DealerRules     01006           WflServer
SomeBank       Charlotte       01011           WflServer
```


Tip Enter workspace and queue names instead of * * to see reports on specific tables. For example, to see a report on the table in the Charlotte queue, enter:

table SomeBank WQM001Q001011

Mapping queue column names to field names

The user-defined column names in a WorkFlo queue table are UF000, UF001, UF002, and so on.

In addition to the queue column names, you need to know:

- The queue field names
- How to map the queue field names to the names you need to enter in an SQL select statement

To map the queue field names to column names, see the table [**“Contents of wqs_fields” on page 100.**](#)

To identify the queue field names, use the WQS_tool DESCQUE command in this format:

DESCQUE <workspace> <queue name>

For example, to display the queue field names and the corresponding database column names for the Dist1 queue in the workspace called workQs, enter:

DESCQUE workQs Dist1

This displays a report similar to the following:

```
<WQS_tool>DESCQUE workQs Dist1
```

Queue:workQs/Dist1 Server: WflServer:corona:FileNet
Table id:01028

| Field | DB Column | Indexed? | Unique? |
|------------|-----------|----------|---------|
| ----- | ----- | ----- | ----- |
| DocumentID | UF000 | | |
| name | UF001 | | |

Saving WQS_tool commands and output to a file

Use these statements in WQS_tool to save both your commands and output to a file:

```
outputfile <filename>  
output on
```

For example, enter:

```
outputfile /tmp/WQ_info  
  
output on  
  
table * *
```

This command creates the file /tmp/WQ_info containing the table statement and the output of the table statement. All subsequent statements and output are appended to the output file until you enter this command to stop:

```
output off
```

Using spacerpt to view tables

By default, the **spacerpt** tool reports on all FileNet tables in the RDBMS database.

By using the `-u` option with a username, you can restrict the tables that display.

To see all tables, enter:

```
spacerpt -u f_sw
```

Creating views

Use **dclview** to create a view for each document class and a general view of all columns in doctaba. Creating a view consumes very little disk space.

The name of a document class view is `F_<document class name>`, and the name of the general view is `F_DOCTABA`. A document class view contains all the FileNet columns and the user columns for the particular document class.

These views translate column names to index names, integer dates to actual dates, and so on. In other words, **dclview** produces a user-friendly view of the database. You can query on a view using an SQL select statement like you can for any other table.

The syntax for the **dclview** command is:

```
dclview [-c] [-g] [-a]<docclass1name> [<docclass2 docclass3 ...>]
```

The following table describes the options you can use with the dclview command.

| Option | Action |
|-------------|--|
| -c | Creates a view in the database. Omit the -c option to display the CREATE VIEW statements on the standard output without creating a view. |
| <doc-class> | Creates a view for one or more specified document classes. |
| -g | Creates the general view on all columns. Can be combined with <docclass>. |
| -a | Creates views on the general view and on all document classes. |

The dclview program creates public synonyms for each view. The columns of a view are different from the equivalent columns in doctaba in the following ways:

- F_DOCCLASSNUMBER is replaced by F_DOCCLASSNAME
- All date columns, which are integers in doctaba, are converted to actual dates and displayed using the default format DD-MON-YYYY
- F_DOCTYPE displays Image, Text, Form, Mixed, or Other instead of a numeric value
- F_RETENTBASE displays as Closing or Entry instead of a numeric value
- F_RETENTDISP displays as Delete or Archive instead of a numeric value

- F_PAGES displays as 1 instead of NULL for single page documents
- F_ACCESSRIGHTS automatically displays as hexadecimal bytes

User column names are distinguished by case. As a result, whenever you use any column names with lowercase letters, you must surround those letters with double quotes (for example, "abc").

You can list a view in the same way that you list a table. The described method lists views you created using dclview and views created for you by the FileNet software. To list views, use this select statement:

```
select owner, view_name from all_views where owner='F_SW';
```

Creating WorkFlo Queue tables and views

Creating a new WorkFlo queue creates both a table and a view on the table. On an existing WorkFlo queue, the software creates the view when a user opens the queue. The name of such a view is:

workspace.queue

Use the view to refer to a queue table without using the queue ID.

A view on a WorkFlo queue:

- Translates F_PRITIME to two fields, F_PRIORITY and F_ENTRYTIME
- Translates dates from integers to actual dates using the Oracle default date format DD-MON-YYYY

Tip

F_ENTRYTIME, which is not translated, is the number of seconds from midnight on January 1, 1970.

Determining maximum string lengths

String lengths can be helpful in planning the layout of a report. To determine the maximum length of all string fields in doctaba, use this select statement:

```
select F_INDEXNAME, F_MAXIXSIZE from USER_INDEX  
where F_INDEXTYPE='2';
```

Example output:

| F_INDEXNAME | F_MAXIXSIZE |
|----------------|-------------|
| ----- | ----- |
| F_DOCTYPE | 1 |
| F_RETENTBASE | 1 |
| F_RETENTDISP | 1 |
| F_ACCESSRIGHTS | 12 |
| test1 | 3 |
| test2 | 2 |

6 rows selected.

To determine the maximum length of a particular string field in doctaba, use this select statement:

```
select F_MAXIXSIZE from USER_INDEX where F_INDEX-  
NAME='<name>';
```

Replace <name> with the case-sensitive index name. For index types other than string, F_MAXIXSIZE is always 0, indicating that no maximum exists.

Working with dates

FileNet software stores a date as an integer. A date is the number of days from the fixed date January 1, 1970. For example, January 4, 1970 is stored as the value 3.

As explained below, you can use dclview to create a view that translates integer dates to actual dates for document records. To query on folder dates or WorkFlo queue dates, however, you need to translate an integer date to an actual date using the **to_date** function as shown in the examples below. Optionally, you can change the output date format using the **to_char** function as well. See the *SQL Language Reference Manual* for more information on dates and functions.

Translating an integer date to a date

In this example, the to_date function translates the document entry date into the default Oracle date format.

```
select F_DOCNUMBER, to_char(to_date('1/1/1970','mm/dd/yyyy') +F_ENTRYDATE, 'DD/MM/YYYY') "date" from DOCTABA;
```

The output looks like this:

| doc# | date |
|------------|------------|
| ----- | ----- |
| 2100004324 | 19/05/1997 |
| 2100004323 | 05/03/2001 |
| 2100004322 | 23/01/2001 |
| 2100004320 | 23/01/2001 |
| 2100004307 | 05/01/2001 |
| 2100004030 | 19/12/2000 |

Changing the Date Format

This example produces the same list as the previous example but uses the `to_char` function to change the output date format and renames the date expression column.

```
select F_DOCNUMBER "doc#",to_char(to_date('1/1/1970','mm/  
dd/yyyy')+F_ENTRYDATE,'fmMonth ddth,YyyY') "date" from  
DOCTABA;
```

The output:

| doc# | date |
|------------|----------------------|
| ----- | ----- |
| 2100004030 | December, 19th, 2000 |
| 2100004307 | January, 5th, 2001 |
| 2100004320 | January, 23rd, 2001 |
| 2100004321 | January, 23rd, 2001 |
| 2100004322 | January, 23rd, 2001 |
| 2100004323 | March, 5th, 2001 |
| 2100004324 | May, 19th, 1997 |

Selecting documents based on dates

This SQL statement selects documents based on dates. Subtracting the fixed date (0) from the desired date (an integer) yields a number of days that matches the data in the database.

```
select F_DOCNUMBER, F_ENTRYDATE, to_char(to_date('1/1/1970','mm/dd/yyyy')+f_entrydate) from DOCTABA where to_char(to_date('1/1/1970','mm/dd/yyyy')+F_ENTRYDATE)=(‘05-JAN-01’);
```

The output:

| F_DOCNUMBER | F_ENTRYDATE | TO_CHAR (TO_DATE ('1/1/1970' , 'mm/dd/yyyy') +F_EN... |
|-------------|-------------|---|
| 2100004304 | 11327 | 05-JAN-01 |
| 2100004305 | 11327 | 05-JAN-01 |
| 2100004306 | 11327 | 05-JAN-01 |
| 2100004307 | 11327 | 05-JAN-01 |

Working with Microsoft SQL Server

This chapter explains the following topics:

- How to use osql statements and WQS_tool to find the index table and column names for use in queries (We assume that you know isql.)
- How to save WQS_tool output to a file
- How to use dclview to create views, which give you a more user-friendly view of document index values
- How to use WorkFlo queue views, which are created for you automatically
- How to find the size of a field

Using isql

If you use Microsoft SQL Server, you can send isql statements to your RDBMS. Enter isql statements, which are case sensitive, exactly as shown in the examples.

Changing the password

To change the f_sqi password, enter this series of commands:

```
sp_password “<oldpassword>”, “<newpassword>”
```

```
go
```

Listing tables

A set of standard FileNet tables always exists in the index database. Table names beginning with wqm are WorkFlo queues. See [Chapter 5, “Table descriptions,” on page 70](#).

To list the names of all FileNet-created tables in the index database, enter this series of commands on the Index server:

```
use fnsys
```

```
go
```

```
select user_name(uid), name from sysobjects where type='U'  
and (uid=user_id('f_sw') or uid=user_id('f_sqi'))
```

```
go
```

This select statement produces a list similar to the following (on a WorkFlo Queue server, this statement lists only WorkFlo queues):

```

name
-----
f_sw          sys_numbers
f_sw          user_index
f_sw          index_cluster
f_sw          document_class
f_sw          doc_class_index
f_sw          doctaba
f_sw          folder
f_sw          folder_contents
f_sw          folder_tabs
f_sw          no_cat_audit
f_sw          menu
f_sw          menu_items
f_sw          validation_tab
f_sw          validation_tab_items
f_sw          wqs_idseed
f_sw          wqs_workspaces
f_sw          wqs_queues
f_sw          wqs_fields
(19 rows affected)

```

Listing the columns of a table

Users logged in as f_sqi can explicitly list the FileNet tables. Users logged in as f_maint can use sys.dbc_tab_columns. To list a table's columns, enter an isql select statement:

```
select name from syscolumns where id=object_id('<table>')
```

Replace <table> with the table name using lowercase letters. When you use a select statement to list the columns of the doctaba table, you see something like the following:

```
name
-----
f_docnumber
f_docclassnumber
f_entrydate
f_lastaccess
f_annotationflag
f_archivedate
f_purgedate
f_deletedate
f_retentbase
f_retentdisp
f_retentoffset
f_pages
f_securityspec
f_accessrights
f_doctype
f_status
f_docformat
f_doclocation
a31
a32
a33
a34
a35
a36
(24 rows affected)
```

If you use Microsoft SQL Server, you can list the columns of a table with an isql statement in this format:

```
sp_help '<table>'
```

Replace <table> with the name of the desired table. For example, to list information for doctaba, enter this series of commands:

sp_help 'f_sw.doctaba'

go

The detailed output lists the name of each column in the table, whether the column requires a value (not null), and the column type.

| Name | Owner | Type | Data_located_on_segment | | | | When_created | |
|------------------|---|------------|-------------------------|-------|-------------------------------|--------------|-------------------------|----------|
| ----- | ----- | ----- | ----- | | | | ----- | |
| doctaba | f_sw | user table | default | | | | Jan 31 1997 11:12AM | |
| Column_name | Type | Length | Prec | Scale | Nulls | Default_name | Rule_name | Identity |
| ----- | ----- | ----- | ----- | ----- | ----- | ----- | ----- | ----- |
| f_docnumber | numeric | 6 | 10 | 0 | 0 | NULL | NULL | 0 |
| f_docclassnumber | int | 4 | NULL | NULL | 0 | NULL | NULL | 0 |
| f_entrydate | int | 4 | NULL | NULL | 0 | NULL | NULL | 0 |
| . | | | | | | | | |
| . | | | | | | | | |
| . | | | | | | | | |
| a31 | numeric | 7 | 14 | 5 | 1 | NULL | NULL | 0 |
| a32 | numeric | 8 | 15 | 7 | 1 | NULL | NULL | 0 |
| a33 | numeric | 8 | 16 | 7 | 1 | NULL | NULL | 0 |
| index_name | index_description | | | | index_keys | | index_max_rows_per_page | |
| ----- | ----- | | | | ----- | | ----- | |
| da_docnumber | clustered, unique located on default | | | | f_docnumber | | 69 | |
| da_archivedate | nonclustered, unique located on default | | | | f_archivedate, f_docnumber | | 89 | |
| da_deletedate | nonclustered unique located on default | | | | f_deletedate f_docnumber | | 87 | |
| da_a49 | nonclustered unique located on default a49 | | | | f_docnumber | | 71 | |

Viewing user index names

In doctaba, the user-defined indexes are named a31, a32, and so on. To view numbered column names as the actual index names, enter:

```
select f_columnname, f_indexname from f_sw.user_index  
  
go
```

This select statement, which produces a list similar to the one shown below, maps the doctaba column names a41, a42, and so on, to the user-defined index names stored in the user_index table.

```
f_columnname f_indexname
-----
NULL          F_DOCNUMBER
NULL          F_DOCCLASSNUMBER
NULL          F_ARCHIVEDATE
NULL          F_PURGEDATE
NULL          F_DELETEDATE
NULL          F_ENTRYDATE
NULL          F_LASTACCESS
NULL          F_RETENTOFFSET
NULL          F_PAGES
NULL          F_DOCTYPE
NULL          F_RETENTBASE
NULL          F_RETENTDISP
NULL          F_ACCESSRIGHTS
NULL          F_DOCFORMAT
NULL          F_DOCLOCATION
a40           User_index_num
a41           DVT_num_idx1
a42           DVT_ascii_idx1
a43           DVT_date_idx1
a44           DVT_menu_idx1
a45           DVT_num_idx2
a46           DVT_ascii_idx2
a47           DVT_date_idx2
a48           DVT_menu_idx2
(24 rows affected)
```

Managing WorkFlo queues

A WQS_tool table report gives you the names of the workspaces and queues on the server, including the server ID and table ID that relate to the queue name. The name format is **wqmsssqqnnnnnn**, where **sss** is the server ID and **nnnnnn** is the table ID. For example, the table name of the Charlotte queue in the report below is **wqm001q001011**.

To enter the WQS_tool utility for managing WorkFlo queues, enter the following command at the operating system prompt:

WQS_tool

To list the workspaces, queue names, and table IDs, enter:

table * *

This displays a report:

| Workspace | Queue name | table name | Queue Server |
|-----------|-------------|------------|--------------|
| ----- | ----- | ----- | ----- |
| pcwfl | ApplDistQ | 01001 | WflServer |
| SomeBank | Fax_In | 01007 | WflServer |
| SomeBank | Greensboro | 01010 | WflServer |
| SomeBank | DealerRules | 01006 | WflServer |
| SomeBank | Charlotte | 01011 | WflServer |

Note Enter workspace and queue names instead of * * to see reports on specific tables. For example, to see a report on the table in the Charlotte queue, enter:

table SomeBank wqm001q001011

Mapping queue column names to field names

The user-defined column names in a WorkFlo queue table are uf000, uf001, uf002, and so on. In addition to the queue column names, you need to know:

- the queue field names
- how to map the queue field names to the names you need to enter in an isql statement

To map the queue field names to column names, see [“Contents of wqs_fields” on page 100](#).

To identify the queue field names, use the WQS_tool DESCQUE command in this format:

DESCQUE <workspace> <queue name>

For example, to display the queue field names and the corresponding database column names for the Dist1 queue in the workspace called workQs, enter:

DESCQUE workQs Dist1

This displays a report similar to the following:

| | | | |
|---|-----------|----------|---------|
| Queue:workQs/Dist1 Server: WflServer:corona:FileNet | | | |
| Table id: 01028 | | | |
| Field | DB Column | Indexed? | Unique? |
| ----- | ----- | ----- | ----- |
| DocumentID | UF000 | | |
| name | UF001 | | |

Saving WQS_tool commands and output to a file

Use these statements in WQS_tool to save both your commands and output to a file:

```
outputfile <filename>  
output on
```

For example, enter:

```
outputfile /tmp/WQ_info  
  
output on  
  
table * *
```

This command creates the file /tmp/WQ_info containing the table statement and the output of the table statement. All subsequent statements and output are appended to the output file until you enter this command to stop:

```
output off
```

Using spacerpt to view tables

By default, the **spacerpt** tool reports on all FileNet tables in the RDBMS database.

```
spacerpt
```

Creating views

Use **dclview** to create a view for each document class and a general view of all columns in doctaba. Creating a view consumes very little disk space.

The name of a document class view is f_<document class name>, and the name of the general view is f_doctaba. A document class view contains all the FileNet columns and the user columns for the particular document class.

These views translate column names to index names, integer dates to actual dates, and so on. In other words, dclview produces a user-friendly view of the database. You can query on a view using an isql statement like you can for any other table.

The syntax for the dclview command is:

```
dclview [-c] [-g] [-a]<docclass1 name> [<docclass2 docclass3 ...>]
```

The following table describes the options you can use with the dclview command.

| Option | Action |
|-------------|--|
| -c | Creates a view in the database. Omit the -c option to display the CREATE VIEW statements on the standard output without creating a view. |
| <doc-class> | Creates a view for one or more specified document classes. |

| Option | Action |
|--------|---|
| -g | Creates the general view on all columns. Can be combined with <docclass>. |
| -a | Creates views on the general view and on all document classes. |

The columns of a view are different from the equivalent columns in doctaba in the following ways:

- f_docclassnumber is replaced by f_docclassname.
- f_pages displays as 1 instead of null for single page documents.

The views are created with case-sensitive column names, because user column names are distinguished by case.

You can list a view in the same way that you list a table. The described method lists views you created using dclview and views created for you by the FileNet software.

To list views, enter the following series of commands:

```
use fnsys
```

```
go
```

```
select user_name(uid), name from sysobjects where type='v'  
and uid=user_id('f_sw')
```

```
go
```

Creating WorkFlo queue tables and views

Creating a new WorkFlo queue creates both a table and a view on the table. On an existing WorkFlo queue, the software creates the view when a user opens the queue.

The name of such a view is:

```
<workspace>.<queue>
```

Use the view to refer to a queue table without using the queue ID.

A view on a WorkFlo queue translates f_pritime to two fields, f_priority and f_entrytime.

Determining maximum string lengths

String lengths can be helpful in planning the layout of a report. To determine the maximum length of all string fields in doctaba, use this isql select statement:

```
select f_indexname, f_maxixsize from f_sw.user_index where
f_indextype='2'

go
```

Example output:

| f_indexname | f_maxixsize |
|--------------------|-------------|
| ----- | ----- |
| F_DOCTYPE | 1.00000 |
| F_RETENTBASE | 1.00000 |
| F_RETENTDISP | 1.00000 |
| F_ACCESSRIGHTS | 12.00000 |
| F_DOCFORMAT | 239.00000 |
| F_DOCLOCATION | 239.00000 |
| DVT_ascii_idx1 | 50.00000 |
| DVT_ascii_idx2 | 50.00000 |
| DVT_str_cluster | 50.00000 |
| PO_NUMBER | 25.00000 |
| SAP_ID | 15.00000 |
| DOC_TYPE | 5.00000 |
| INVOICE_NUMBER | 20.00000 |
| ACK_NUMBER | 10.00000 |
| CREDIT_MEMO_NUMBER | 20.00000 |
| (15 rows affected) | |

To determine the maximum length of a particular string field in doc-taba, use this isql select statement:

```
select f_maxixsize from user_index where f_index-  
name='<name>'
```

```
go
```

Replace <name> with the case sensitive index name.

For index types other than string, f_maxixsize is always 0, indicating that no maximum exists.

4

Working with DB2

This chapter explains the following topics:

- How to use access DB2, logon to the database instance, and work with the tables
- How to manage WorkFlo Queues
- How to create views
- How to determine string lengths
- How to work with dates

Using DB2

Depending upon the server platform you complete one of the following steps to access DB2:

- On UNIX, enter db2 at the command line to start the DB2 Command Line Processor.
- On Windows Server, select the program IBM DB2, then select Command Line Tools, then select Command Line Center or Command Line Processor.

Logon to the database instance

When you are in the Command Line processor, enter the following:

connect to <database instance name> user f_sw using f_sw

You will then see a display similar to the following::

```
db2 => connect to indexdb user f_sw using f_sw
```

Database Connection Information

```
Database Server          = DB2/AIX64 8.1.0
SQL authorization ID     = F_SW
Local database alias    = INDEXDB
```

Listing tables

A set of standard FileNet tables always exists in the index database. Table names beginning with wqm are WorkFlo queues. See [Chapter 5, “Table descriptions,” on page 70](#).

Tip

The tables in this section are owned by F_SW.

To list the names of all FileNet-created tables in the index database, enter the following :

list tables

This select statement produces a list similar to the following:

```
db2 => list tables
```

| Table/View | Schema | Type | Creation time |
|----------------------|--------|------|----------------------------|
| DOC_CLASS_INDEX | F_SW | T | 2003-12-02-09.33.56.465145 |
| DOCTABA | F_SW | T | 2003-12-02-09.33.57.572030 |
| DOCUMENT_CLASS | F_SW | T | 2003-12-02-09.33.55.276572 |
| F_DOCTABA | F_SW | T | 2003-12-04-15.26.57.229072 |
| F_MSAR_1GB | F_SW | T | 2003-12-04-15.26.58.504508 |
| FOLDER | F_SW | T | 2003-12-02-09.33.58.910654 |
| FOLDER_CONTENTS | F_SW | T | 2003-12-02-09.34.00.279018 |
| FOLDER_TABS | F_SW | T | 2003-12-02-09.34.01.558285 |
| GUIDS | F_SW | T | 2003-12-02-09.33.53.026819 |
| INDEX_CLUSTER | F_SW | T | 2003-12-02-09.33.54.180740 |
| MENU | F_SW | T | 2003-12-02-09.34.03.548612 |
| MENU_ITEMS | F_SW | T | 2003-12-02-09.34.05.163902 |
| NO_CAT_AUDIT | F_SW | T | 2003-12-02-09.34.02.603993 |
| SYS_NUMBERS | F_SW | T | 2003-12-02-09.33.50.810387 |
| USER_INDEX | F_SW | T | 2003-12-02-09.33.51.831782 |
| VALIDATION_TAB | F_SW | T | 2003-12-02-09.34.06.219875 |
| VALIDATION_TAB_ITEMS | F_SW | T | 2003-12-02-09.34.07.391230 |
| WQM001Q000001 | F_SW | T | 2003-12-04-11.53.16.538700 |
| WQM001Q000002 | F_SW | T | 2003-12-05-09.59.36.152675 |
| WQS_FIELDS | F_SW | T | 2003-12-04-11.15.54.243849 |
| WQS_IDSEED | F_SW | T | 2003-12-04-11.15.53.609280 |
| WQS_QUEUES | F_SW | T | 2003-12-04-11.15.53.937636 |
| WQS_RELEASE | F_SW | T | 2003-12-04-11.15.54.498134 |
| WQS_WORKSPACES | F_SW | T | 2003-12-04-11.15.53.700585 |

24 records(s) selected.

```
db2 =>
```

Note: The “T” under Type means table, “V” means view.

You can also list all of the tables that belong to F_SW by entering the following:

```
select table_name, table_type from sysibm where table_  
schema = 'F_SW'
```

This select statement produces a list similar to the following:

```
db2 => select table_name, table_type from sysibm.tables where table_
schema = 'F_SW'
```

| TABLE_NAME | TABLE_TYPE |
|----------------------|------------|
| ----- | ----- |
| DOC_CLASS_INDEX | BASE TABLE |
| DOCTABA | BASE TABLE |
| DOCUMENT_CLASS | BASE TABLE |
| F_DOCTABA | VIEW |
| F_MSAR_1GB | VIEW |
| FOLDER | BASE TABLE |
| FOLDER_CONTENTS | BASE TABLE |
| FOLDER_TABS | BASE TABLE |
| GUIDS | BASE TABLE |
| INDEX_CLUSTER | BASE TABLE |
| MENU | BASE TABLE |
| MENU_ITEMS | BASE TABLE |
| NO_CAT_AUDIT | BASE TABLE |
| SYS_NUMBERS | BASE TABLE |
| USER_INDEX | BASE TABLE |
| VALIDATION_TAB | BASE TABLE |
| VALIDATION_TAB_ITEMS | BASE TABLE |
| WQM001Q000001 | BASE TABLE |
| WQM001Q000002 | BASE TABLE |
| WQS_FIELDS | BASE TABLE |
| WQS_IDSEED | BASE TABLE |
| WQS_QUEUES | BASE TABLE |
| WQS_RELEASE | BASE TABLE |
| WQS_WORKSPACES | BASE TABLE |

24 records(s) selected.

Listing the columns of a table

The following command lists all column definitions of a table:

describe table <tablename>

Replace <tablename> with the table name using lowercase letters.

When you use a describe statement to list the columns of the doctaba table, you see something like the following:

:

db2 => describe table doctaba

| Column name | Type schema | Type name | Length | Scale | Null |
|-------------------------|----------------|--------------|--------|-------|------|
| F_DOCNUMBER | SYSIBM | DECIMAL | 10 | 0 | No |
| F_DOCCLASSNUMBER | SYSIBM | INTEGER | 4 | 0 | No |
| F_ENTRYDATE | SYSIBM | INTEGER | 4 | 0 | No |
| F_LASTACCESS | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_ANNOTATIONFLAG | SYSIBM | VARCHAR | 1 | 0 | Yes |
| F_ARCHIVEDATE | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_PURGEDATE | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_DELETEDATE | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_RETENTBASE | SYSIBM | VARCHAR | 1 | 0 | Yes |
| F_RETENTDISP | SYSIBM | VARCHAR | 1 | 0 | Yes |
| F_RETENTOFFSET | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_PAGES | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_SECURITYSPEC | SYSIBM | VARCHAR | 12 | 0 | Yes |
| F_ACCESSRIGHTS | SYSIBM | VARCHAR | 12 | 0 | Yes |
| F_DOCTYPE | SYSIBM | VARCHAR | 1 | 0 | Yes |
| F_STATUS | SYSIBM | INTEGER | 4 | 0 | Yes |
| F_DOCFORMAT | SYSIBM | VARCHAR | 239 | 0 | Yes |
| F_DOCLOCATION | SYSIBM | VARCHAR | 239 | 0 | Yes |
| A31 | SYSIBM | DECIMAL | 18 | 4 | Yes |
| A32 | SYSIBM | VARCHAR | 239 | 0 | Yes |
| A33 | SYSIBM | INTEGER | 4 | 0 | Yes |
| A34 | SYSIBM | VARCHAR | 14 | 0 | Yes |
| A35 | SYSIBM | DECIMAL | 18 | 4 | Yes |
| A36 | SYSIBM | VARCHAR | 239 | 0 | Yes |
| A37 | SYSIBM | INTEGER | 4 | 0 | Yes |
| A38 | SYSIBM | VARCHAR | 14 | 0 | Yes |
| A39 | SYSIBM | DECIMAL | 18 | 4 | Yes |
| A40 | SYSIBM | VARCHAR | 239 | 0 | Yes |
| 28 records(s) selected. | | | | | |

You can also list the column names of a table by entering the following select statement:

```
select column_name from sysibm.columns where table_name  
= 'DOCTABA'
```

This select statement produces a list similar to the following:

```
db2 => select column_name from sysibm.columns where table_name = 'DOCTABA'
```

```
COLUMN_NAME
```

```
-----  
A31  
A32  
A33  
A34  
A35  
A36  
A37  
A38  
A39  
A40  
F_ACCESSRIGHTS  
F_ANNOTATIONFLAG  
F_ARCHIVEDATE  
F_DELETEDATE  
F_DOCCLASSNUMBER  
F_DOCFORMAT  
F_DOCLOCATION  
F_DOCNUMBER  
F_DOCTYPE  
F_ENTRYDATE  
F_LASTACCESS  
F_PAGES  
F_PURGEDATE  
F_RETENTBASE  
F_RETENTDISP  
F_RETENTOFFSET  
F_SECURITYSPEC  
F_STATUS
```

```
28 records(s) selected.
```

Viewing user index names

In doctaba, the user-defined indexes are named A31, A32, and so on. To view numbered column names as the actual index names, enter the following:

```
select f_columnname, f_indexname from user_index
```

This statement produces a list similar to the one shown below. It maps the doctaba column names A31, A32, and so on, to the user-defined index names stored in the user_index table.

```
db2 => select f_columnname, f_indexname from user_index
```

| F_COLUMN_NAME | F_INDEXNAME |
|---------------|------------------|
| ----- | ----- |
| - | F_DOCNUMBER |
| - | F_DOCCLASSNUMBER |
| - | F_ARCHIVEDATE |
| - | F_PURGEDATE |
| - | F_DELETEDATE |
| - | F_ENTRYDATE |
| - | F_LASTACCESS |
| - | F_RETENTOFFSET |
| - | F_PAGES |
| - | F_DOCTYPE |
| - | F_RETENTBASE |
| - | F_RETENTDISP |
| - | F_ACCESSRIGHTS |
| - | F_DOCFORMAT |
| - | F_DOCLOCATION |
| a31 | DVT_num_idx1 |
| a32 | DVT_acsii_idx1 |
| a33 | DVT_date_idx1 |
| a34 | DVT_menu_idx1 |
| a35 | DVT_num_idx2 |
| a36 | DVT_ascii_idx2 |
| a37 | DVT_date_idx2 |
| a38 | DVT_menu_idx2 |
| a39 | DVT_num_cluster |
| a40 | DVT_str_cluster |

25 records(s) selected.

```
db2 =>
```

Managing WorkFlo queues

A WQS_tool table report gives you the names of the workspaces and queues on the server, including the server ID and table ID that relate to the queue name. The name format is **WQMsssQnnnnnn**, where **sss** is the server ID and **nnnnnn** is the table ID. For example, the table name of the Charlotte queue in the report below is **WQM001Q001011**.

If this is a fresh install, all queues should have **WQMsssQnnnnnn** as the name format. If existing customers have moved from the Oracle or MSSQL database to the DB2 database, some older queues that were created under IMS 3.0.3 or earlier can have **WQM1nnnnnn** as the format, where **nnnnnn** is the table ID listed in the report.

To enter the WQS_tool utility for managing WorkFlo queues, enter the following command at the operating system prompt:

WQS_tool

To list the workspaces, queue names, and table IDs, enter:

table * *

This displays a report:

| <WQS_tool>table * * | | | |
|---------------------|-------------|------------|--------------|
| Workspace | Queue name | table name | Queue Server |
| ----- | ----- | ----- | ----- |
| pcwfl | ApplDistQ | 01001 | WflServer |
| SomeBank | Fax_In | 01007 | WflServer |
| SomeBank | Greensboro | 01010 | WflServer |
| SomeBank | DealerRules | 01006 | WflServer |
| SomeBank | Charlotte | 01011 | WflServer |

Note Enter workspace and queue names instead of * * to see reports on specific tables. For example, to see a report on the table in the Charlotte queue, enter:

table SomeBank WQM001Q001011

Mapping queue column names to field names

The user-defined column names in a WorkFlo queue table are UF000, UF001, UF002, and so on.

In addition to the queue column names, you need to know:

- The queue field names
- How to map the queue field names to the names you need to enter in an SQL select statement

To map the queue field names to column names, see the table [**“Contents of wqs_fields” on page 100.**](#)

To identify the queue field names, use the WQS_tool DESCQUE command in this format:

DESCQUE <workspace> <queue name>

For example, to display the queue field names and the corresponding database column names for the Dist1 queue in the workspace called workQs, enter:

DESCQUE workQs Dist1

This displays a report similar to the following:

```
<WQS_tool>DESCQUE workQs Dist1
```

Queue:workQs/Dist1 Server: WflServer:corona:FileNet
Table id:01028

| Field | DB Column | Indexed? | Unique? |
|------------|-----------|----------|---------|
| ----- | ----- | ----- | ----- |
| DocumentID | UF000 | | |
| name | UF001 | | |

Saving WQS_tool commands and output to a file

Use these statements in WQS_tool to save both your commands and output to a file:

```
outputfile <filename>  
output on
```

For example, enter:

```
outputfile /tmp/WQ_info  
  
output on  
  
table * *
```

This command creates the file /tmp/WQ_info containing the table statement and the output of the table statement. All subsequent statements and output are appended to the output file until you enter this command to stop:

```
output off
```


Creating views

Use **dclview** to create a view for each document class and a general view of all columns in doctaba. Creating a view consumes very little disk space.

The name of a document class view is `f_<document class name>`, and the name of the general view is `f_doctaba`. A document class view contains all the FileNet columns and the user columns for the particular document class.

These views translate column names to index names, integer dates to actual dates, and so on. In other words, `dclview` produces a user-friendly view of the database. You can query on a view using an `isql` statement like you can for any other table.

The syntax for the `dclview` command is:

```
dclview [-c] [-g] [-a]<docclass1name> [<docclass2 docclass3 ...>]
```

The following table describes the options you can use with the `dclview` command.

| Option | Action |
|--------------------------------|--|
| <code>-c</code> | Creates a view in the database. Omit the <code>-c</code> option to display the <code>CREATE VIEW</code> statements on the standard output without creating a view. |
| <code><doc-class></code> | Creates a view for one or more specified document classes. |
| <code>-g</code> | Creates the general view on all columns. Can be combined with <code><docclass></code> . |
| <code>-a</code> | Creates views on the general view and on all document classes. |

The columns of a view are different from the equivalent columns in doctaba in the following ways:

- f_docclassnumber is replaced by f_docclassname.
- f_pages displays as 1 instead of null for single page documents.

The views are created with case-sensitive column names, because user column names are distinguished by case.

You can list a view in the same way that you list a table. The described method lists views you created using dclview and views created for you by the FileNet software.

To list views, enter the following series of commands:

**select name, qualifier from sysibm.sysviews where creator =
'F_SW'**

```
db2 => select name, qualifier from sysibm.sysviews where creator = 'F_SW'
```

| NAME | Qualifier |
|------------|-----------|
| ----- | ----- |
| F_DOCTABA | F_SW |
| F_MSAR_1GB | F_SW |

Creating WorkFlo queue tables and views

Creating a new WorkFlo queue creates both a table and a view on the table. On an existing WorkFlo queue, the software creates the view when a user opens the queue.

The name of such a view is:

`<workspace>.<queue>`

Use the view to refer to a queue table without using the queue ID.

A view on a WorkFlo queue translates `f_pritime` to two fields, `f_priority` and `f_entrytime`.

Determining maximum string lengths

String lengths can be helpful in planning the layout of a report. To determine the maximum length of all string fields in doctaba, use this select statement:

```
select F_INDEXNAME, F_MAXIXSIZE from USER_INDEX
where F_INDEXTYPE='2';
```

Example output:

| F_INDEXNAME | F_MAXIXSIZE |
|----------------|-------------|
| ----- | ----- |
| F_DOCTYPE | 1 |
| F_RETENTBASE | 1 |
| F_RETENTDISP | 1 |
| F_ACCESSRIGHTS | 12 |
| test1 | 3 |
| test2 | 2 |

6 rows selected.

To determine the maximum length of a particular string field in doctaba, use this select statement:

```
select F_MAXIXSIZE from USER_INDEX where F_INDEX-  
NAME='<name>;
```

Replace <name> with the case-sensitive index name. For index types other than string, F_MAXIXSIZE is always 0, indicating that no maximum exists.

Working with dates

FileNet software stores a date as an integer. A date is the number of days from the fixed date January 1, 1970. For example, January 4, 1970 is stored as the value 3. As explained below, you can use dclview to create a view that translates integer dates to actual dates for document records. To query on folder dates or WorkFlo queue dates, however, you need to translate an integer date to an actual date.

Translating an integer date to a date

Enter the following select statement to translate the F_entrydate into a readable format:

```
select f_docnumber, DATE(F_entrydate + 719163) from doctaba
```

The output looks like this:

```
db2 => select f_docnumber, DATE(F_entrydate + 719163) from doctaba

F_DOCNUMBER 2
-----
    100000. 12/02/2003
    100001. 12/02/2003
    100002. 12/02/2003

  3 record(s) selected.

db2 =>
```

Tip: FileNet date begins on 01/01/1970 and DB2 begins its count on 01/01/0001, so you have to add 719163 to the FileNet date for it to correspond to the DB2 date.

Table descriptions

This chapter describes the FileNet tables in the RDBMS databases on a system running the Image Services software.

The index database, which resides on a Combined server or an Index server, is an RDBMS database containing the following FileNet-defined tables:

- Standard tables in the index database
- WQS system tables containing workspace and queue descriptions
- Any number of WorkFlo queue tables (these tables are in the index database only if you define WorkFlo queues)

In addition to WorkFlo queues in the index database, each Application server running WorkFlo Queue services maintains an RDBMS database (the WorkFlo Queue database) containing WorkFlo queues.

Standard tables

These tables appear in every index database:

| | | |
|----------------|-----------------|------------------------|
| ce_id_map | ce_os_dcl_map | doctaba |
| document_class | doc_class_index | export_log |
| folder | folder_contents | folder_tabs (not used) |
| GUIDS | index_cluster | menu |
| menu_items | no_cat_audit | sys_numbers |
| user_index | validation_tab | validation_tab_items |
| wqs_workspaces | wqs_idseed | wqs_queues |
| wqs_fields | | |

ce_id_map table

The ce_id_map table is used to store f_ce_os_id mapping to the object store GUID, the CE domain GUID, the object store name, and the CE domain name.

Contents of ce_id_map

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_CE_OS_ID | f_ce_os_id | The CE Object Store ID that is uniquely assigned by Image Services from the sys_numbers table (page 92). |
| F_CE_OS_GUID | f_ce_os_guid | The CE Object Store GUID assigned by Content Engine. |
| F_CE_DOMAIN_GUID | f_ce_domain_guid | The CE Domain GUID assigned by Content Engine. |
| F_CE_OS_NAME | f_ce_os_name | The CE Object Store name configured by user from Content Engine. |

Contents of ce_id_map, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_CE_DOMAIN_NAME | f_ce_domain_name | The CE Domain name configured by user from Content Engine. |
| F_UPDATE_TIMEDATE | f_update_timedate | An update time/date stamp. |

ce_os_dcl_map table

The columns of the ce_os_dcl_map table associate the CE domain/object store to Image Services document class and it is controlled by Enterprise Manager. More than one object store could be associated (mapped) with one Image Services document class.

Contents of ce_os_dcl_map

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_CE_OS_ID | f_ce_os_id | The CE Object Store ID. |
| F_DCL_ID | f_dcl_id | The Image Services Document Class ID that is associated with the CE Object Store ID. |

doctaba table

The doctaba table has 22 FileNet-defined columns and up to 224 user-defined columns named A31 to A254 (Oracle and DB2) or a31 to a254 (Microsoft SQL Server). Each column represents an index field and each row contains the index values associated with a committed document.

See [“Viewing user index names” on page 23](#) (Oracle users) or [page 41](#) (Microsoft SQL Server users) or [“Viewing user index names” on page 60](#) (DB2 users).

Contents of doctaba

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_DOCNUMBER | f_docnumber | System-assigned document ID number. |
| F_DOCCLASSNUMBER | f_docclassnumber | System-assigned document class number. |
| F_ENTRYDATE | f_entrydate | Date the document was cataloged (the number of days from 1/1/70). |
| F_LASTACCESS | f_lastaccess | Not used. |
| F_ANNOTATIONFLAG | f_annotationflag | Not used. Value is always null. |
| F_ARCHIVEDATE | f_archivedate | Date the document is eligible for archiving (the number of days from 1/1/70). |
| F_PURGEDATE | f_purgedate | Not used. |
| F_DELETEDATE | f_deletedate | Date the document is eligible for deletion (the number of days from 1/1/70). |
| F_RETENTBASE | f_retentbase | Date on which the retention period begins for a given document: null = document close date 1 = document file date (entry date) |

Contents of doctaba, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_RETENTDISP | f_retentdisp | Action to be taken with a document once its retention period ends: null = delete 1 = archive |
| F_RETENTOFFSET | f_retentoffset | Counting from F_RETENTBASE (Oracle) or f_retentbase (Microsoft SQL Server), the number of months until the document is eligible for deletion or archiving. |
| F_PAGES | f_pages | Number of pages in the document: null = 1 page |
| F_SECURITYSPEC | f_securityspec | Not used. |
| F_ACCESSRIGHTS | f_accessrights | Security clearance needed for a given document. |
| F_DOCTYPE | f_doctype | Specifies the document type: null = image 1 = text 2 = form 3 = mixed (more than one type) 4 = not used 5 = other 6 = Document from P8 |
| F_DOCFORMAT | f_docformat | Supports heterogeneous objects as a document type. |
| F_DOCLOCATION | f_doclocation | References externally stored documents. |
| F_STATUS | f_status | Not used. |
| F_ACCESSRIGHTS_RD | f_accessrights_rd | “Read” security clearance needed for a given document. |

Contents of doctaba, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_ACCESSRIGHTS_WR | f_accessrights_wr | “Write” security clearance needed for a given document. |
| F_ACCESSRIGHTS_AX | f_accessrights_ax | “Append/Execute” security clearance needed for a given document. |
| F_CE_OS_ID | f_ce_os_id | The CE Object Store ID the corresponding document is exported to. |
| A31 - A254 | a31 - a254 | Each of these columns represents a user-defined index field. See “Mapping queue column names to field names” on page 25 (Oracle and DB2 users) or page 44 (Microsoft SQL Server users). |

document_class table

The columns of the document_class table describe the attributes of a document class—except for the index fields (which are described in the doc_class_index table). Each row describes a document class.

Contents of document_class

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_DOCCLASSNUMBER | f_docclassnumber | System-assigned document class number. |
| F_DOCCLASSNAME | f_docclassname | User-assigned document class name. |
| F_DESCR | f_descr | User-assigned document class definition. |

Contents of document_class, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_DOCTABLEID | f_doctableid | Not used. |
| F_PAGES | f_pages | Number of pages per document expected for a given document class: 0 = variable number of pages |
| F_BATCHSIZE | f_batchsize | Number of pages per batch expected for a given document class. |
| F_PRIMARYPATH | f_primarypath | Not used. |
| F_OPTIONALDE | f_optionalde | Listing of the optional verification steps selected by default for the particular document class. No commas between the numbers: 2 = image verify 6 = index verify 7 = batch total verify |
| F_BYPASSINDX | f_bypassindx | Not used. |
| F_TABOUT | f_tabout | Specifies if an operator must use the Execute key to exit from an indexing form (document entry on an IWS/CWS only): null = Execute key required y = Execute, Tab, or Return key |
| F_INDEXINGFORM | f_indexingform | Indexing form name used for a given document class. |
| F_QUERYFORM | f_queryform | Not used. |
| F_EXCEPTIONFLAG | f_exceptionflag | Not used. |
| F_DOCTYPE | f_doctype | Not used. |

Contents of document_class, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_WORKFLOWQUEUE | f_workflowqueue | Name of the WorkFlo (distributor) queue to which all document ID numbers in this document class will be added at committal time. |
| F_WORKFLOSYSTEM | f_workflowsystem | Name of the WorkFlo system that has the WorkFlo queue defined in F_WORKFLOWQUEUE (Oracle) or f_workflowqueue (Microsoft SQL Server). |
| F_FAMILYNUMBER | f_familynumber | Number of the media family associated with a given document class. A value of -1 indicates that clustering is in use. |
| F_FAMILYNAME | f_familyname | Name of the media family associated with a given document class. |
| F_RETENTDISP | f_retentdisp | Action to be taken with a document once its retention period expires: null = delete 1 = archive |
| F_RETENTBASE | f_retentbase | Date the retention period begins for a given document: null = document close date 1 = document file date (entry date) |
| F_RETENTOFFSET | f_retentoffset | Counting from F_RETENTBASE (Oracle) or f_retentbase (Microsoft SQL Server), the number of months until the document is eligible for deletion or archiving. |
| F_ARCHIVEPERIOD | f_archiveperiod | Not used. |
| F_SECURITYSPEC | f_securityspec | Not used. |
| F_CONVERTFLAG | f_convertflag | Not used. |

Contents of document_class, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_ACCESSRIGHTS | f_accessrights | Default security information for a document class. |
| F_STATUSFLAG | f_statusflag | Not used. |
| F_NUMBERPATHS | f_numberpaths | Not used. |
| F_NUMBERINDICES | f_numberindices | Number of user-defined index fields associated with a given document class: -1 = not used 0 = document class with no index n = number of index fields |
| F_NUMBERSCANSECTNS | f_numberscansectns | Not used. |
| F_APERCARDFILE | f_apercardfile | Not used. |
| F_GROUP4 | f_group4 | Not used. |
| F_NOCATALOG | f_nocatalog | Indicates whether cataloging is enabled or disabled for this document class: null = enabled y = disabled |
| F_NUMBERGUIDS | f_numberguids | Indicates the number of globally unique identifiers for the document class. One GUID is added automatically to the GUIDS table. min = 1, max = 10 |
| F_DMA_NAME | f_dma_name | Display name initially copied from the document class name. It can be modified but cannot be left blank. |

Contents of document_class, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_DELAYMIGRATE | f_delaymigrate | Controls when documents in this class are migrated to storage media: -1 = no migration 0 = no delay; migrate immediately n = number of seconds to delay before migrating |
| F_ACCESSRIGHTS_RD | f_accessrights_rd | “Read” security clearance needed for a document class. |
| F_ACCESSRIGHTS_WR | f_accessrights_wr | “Write” security clearance needed for a document class. |
| F_ACCESSRIGHTS_AX | f_accessrights_ax | “Append/Execute” security clearance needed for a document class. |
| F_CE_OS_ID | f_ce_os_id | Document class default CE Object Store ID. This value must be set in the CFS Connector - IS Catalog Export Tool run through the Remote Admin Console. Also, the object store to document class relationship must be established before the Remote Admin Console setting is made. Any documents committed to the document class will now generate an entry in the export_log table (see page 83). |

doc_class_index table

The columns of the doc_class_index table describe the attributes of an index field. Each row represents an index field as it is used in one document class. An index field used in two document classes appears in two different rows.

Contents of doc_class_index

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_COLUMNNAME | f_columnname | Name of the index information column in doctaba (for example, A32 or a32). |
| F_INDEXNAME | f_indexname | User-assigned name for the indexing field. |
| F_DOCCLASSNUMBER | f_docclassnumber | System-assigned document class number. |
| F_BATCHTOTAL | f_batchtotal | Specifies if batch totals can be performed on an associated (numeric) index field: null = no 1 = yes |
| F_VERIFYFLAG | f_verifyflag | Specifies if index verification can be performed on an index field: null = no 1 = yes |
| F_REQDFLAG | f_reqdflag | Specifies if an associated index field is mandatory: null = mandatory 1 = optional |
| F_QUERYMATCH | f_querymatch | Not used. |

Contents of doc_class_index, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_OCRFLAG | f_ocrflag | Indicates how data is entered into an associated index field: null = keyboard 1 = OCR (optical character recognition) 2 = not used |
| F_UNITS | f_units | Specifies unit of measure for OCR input: 1 = inches 2 = millimeters |
| F_FONTFILE | f_fontfile | Specifies the OCR font: 1 = 14 pt alphanumeric 2 = 14 pt numeric 3 = 14 pt Times 4 = OCR-B |
| F_DOCPAGENO | f_docpageno | Not used. This value is always set to 0. |
| F_XOFFSET | f_xoffset | For OCR input only, the horizontal offset in F_UNITS (f_units) from the upper left-hand corner of the document. |
| F_YOFFSET | f_yoffset | For OCR input only, the vertical offset in F_UNITS (f_units) from the upper left-hand corner of the document. |
| F_XLENGTH | f_xlength | For OCR input only, the width of the area in F_UNITS (f_units). |
| F_YHEIGHT | f_yheight | For OCR input only, the height of the area in F_UNITS (f_units). |

Contents of doc_class_index, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_OCRSIZE | f_ocrsize | For OCR input only, the maximum number of characters to be read for a given index field. |
| F_APERCARDIXLOC | f_apercardixloc | <p>For OCR input only, shows the location of bar codes on a page. The bar codes can be in 1 to 5 locations on the page. If you use autoindexing, each string index in a document class can have this field.</p> <p>The format of this value is:</p> <p style="text-align: center;">ccllccllccllccllccll</p> <p>where cc is the starting column number of a bar code and ll is the number of columns.</p> <p>For example, with a bar code in columns 1 through 5 and another in columns 20 through 24, the data in this field would be:</p> <p style="text-align: center;">01052005000000000000</p> |

export_log table

The export_log table holds export log information created as part of a document committal if the Document Class Object Store property (f_ce_os_id) is set. The export_log table is also used for exporting existing doctaba entries to P8 when documents are either deleted or updated. The f_ce_os_id entry is also set in doctaba.

Contents of export_log

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_CE_OS_ID | f_ce_os_id | CE Object Store ID where the corresponding document ID is exported to. |
| F_SEQNUM1 | f_seqnum1 | High water mark sequence number (most significant 32 bits) |
| F_SEQNUM2 | f_seqnum2 | High water mark sequence number (least significant 32 bits) |
| F_DOCNUMBER | f_docnumber | The Image Services Document ID. |
| F_ACTION | f_action | Export action to be taken: 1 : Insert during document committal. 2 : Export using the CFS Connector - IS Catalog Export Tool. 3 : Update when document DIRs are updated. 4 : Delete doctaba after export. 5 : Delete when documents are deleted. |
| F_DCL_ID | f_dcl_id | The Image Services Document Class ID that is associated with the corresponding document ID. |
| F_CAT_IN_DOCTABA | f_cat_in_doctaba | 'Y': Catalog in doctaba 'N': No catalog in doctaba |

Contents of export_log, Continued

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_NEW_DIR | f_new_dir | New DIR (Document Index Record) for all action types. These entries are RAW data. |
| F_OLD_DIR | f_old_dir | Old DIR - is only used for updates action - null okay. These entries are RAW data. |

folder table

The columns of the folder table describe folder attributes. Each row describes one folder.

Contents of folder

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_FOLDERNUMBER | f_foldernumber | System-assigned identification number for this folder. |
| F_FOLDERNAME | f_foldername | User-assigned name for the folder. |
| F_CREATIONDATE | f_creationdate | Date the folder was created (stored as the number of days from 1/1/70). |
| F_ARCHIVEDATE | f_archivedate | Date the folder becomes eligible for archiving (stored as the number of days from 1/1/70). |
| F_DELETEDATE | f_deletedate | Date the folder becomes eligible for deletion (stored as the number of days from 1/1/70). |

contents of folder, Continued

| Oracle/DB2 Column Name | MSSQLServer Column Name | Contents |
|---------------------------|----------------------------|--|
| F_RETENTBASE | f_retentbase | Date on which the retention period begins for a given folder: null = folder close date 1 = folder creation date |
| F_RETENTOFFSET | f_retentoffset | Counting from F_RETENTBASE (Oracle) or f_retentbase (Microsoft SQL Server), the number of months until the folder is eligible for deletion or archiving. |
| F_RETENTDISP | f_retentdisp | Action to be taken with a folder once its retention period ends: null = delete 1 = archive |
| F_AUTODELPERIOD | f_autodelperiod | From date filed, the number of months until a document is eligible for automatic unfiled from the folder. |
| F_ACCESSRIGHTS | f_accessrights | Folder security information. |
| F_ACCESSRIGHTS_RD | f_accessrights_rd | “Read” folder security information. |
| F_ACCESSRIGHTS_WR | f_accessrights_wr | “Write” folder security information. |
| F_ACCESSRIGHTS_AX | f_accessrights_ax | “Append/Execute” folder security information. |

folder_contents table

Each row of the folder_contents table represents a document that is filed in a folder. The columns describe the attributes of a filed document in a folder.

Contents of folder_contents

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_FOLDERNUMBER | f_foldernumber | System-assigned identification number for this folder. |
| F_DOCNUMBER | f_docnumber | Document ID number of a document filed in the folder. |
| F_DOCTABLEID | f_doctableid | Used internally by INX when renumbering F_ORDINAL (Oracle) or f_ordinal (Microsoft SQL Server). |
| F_AUTODELETEDATE | f_autodeletedate | Date (stored as the number of days from 1/1/70) the document is eligible to be automatically unfiled from the folder. 0 = not eligible on any date |
| F_ORDINAL | f_ordinal | Sequential position of this document in the folder. This is the order in which the document was filed. |

GUIDS table

The columns of the GUIDS table describe the attributes of a Globally Unique Identifier's table. GUIDs are DMA-compliant, 16-byte integers used to uniquely identify each element transported over a network. The system ensures unique GUID assignments by automatically generating this integer using an algorithm based on the system's network card MAC address and a format that complies with the specifications provided for the system's platform. Each row in the GUIDS table contains the attribute values of a GUID created by the user.

Contents of GUIDS

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_COLUMNNAME | f_columnname | An inherited value that is populated with the column name from the User Index if the GUID is associated with an index, or null if the GUID is associated with a document class. |
| F_DOCCLASSNUMBER | f_docclassnumber | An inherited value that is populated with the document class number if the GUID is associated with a document class, or null if the GUID is associated with an index . |
| F_GUID | f_guid | Never null, always unique (36 characters). |

index_cluster table

The index_cluster table contains information on documents in any document class that uses clustering. It contains the name of the cluster, the cluster ID number, the index fields associated with the document class, the media family involved, and so on.

Contents of index_cluster

| Oracle/DB2 Column Name | MSSQLServer Column Name | Contents |
|---------------------------|----------------------------|--|
| F_CLUSTERNO | f_clusterno | System-assigned cluster space number. Currently only one index cluster space is supported per system. |
| F_COLUMNNAME | f_columnname | Name of the column in doctaba that contains the value for a specific index field. |
| F_INDEXNAME | f_indexname | User-assigned name of this indexing field. |
| F_FAMILYNAME | f_familyname | Name of the media family on which documents clustered under this index name will be stored. |
| F_FAMILYNO | f_familyno | System-assigned identification number for a given media family name. |
| F_CLUSTERSIZE | f_clustersize | Expected average number of documents for this cluster. |
| F_INDEXTYPE | f_indextype | The type of indexing field: 1 = numeric 2 = string 4 = menu 8 = date |
| F_UPCASE | f_upcase | Case specification for the associated string index value: null = string stored in uppercase 1 = string stored as entered in uppercase, lowercase, or mixed case |

menu table

The columns of the menu table describe the attributes of a menu. Each row contains the attribute values of a user-created menu.

Contents of menu

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_MENUNUMBER | f_menunumber | System-assigned menu ID number. A new number is assigned if the menu is updated. |
| F_MENUNAME | f_menuname | User-assigned menu name. |
| F_DESC | f_desc | User-defined menu description. |
| F_LASTMOD | f_lastmod | The number of seconds since 12:00 a.m. January 1, 1970 that the menu was created or last modified. |
| F_USERCODE | f_usercode | Version information for internal use only. |
| F_NUMITEMS | f_numitems | Number of choices in the menu. |
| F_TRANSLATERULE | f_translaterule | Rule to be followed in translation. |
| F_LANGUAGE | f_language | Language character set for the menu: y = English a = Arabic |

menu_items table

The columns of the menu_items table describe the attributes of a menu item. Each row contains the attribute values of a menu item from a user-created menu.

Contents of menu_items

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_MENUNUMBER | f_menunumber | System-assigned menu ID number. |
| F_ORDINAL | f_ordinal | Sequential position of this item in the menu. |
| F_ITEMCODE | f_itemcode | User-assigned code for this menu item. |
| F_ITEMDESC | f_itemdesc | User-defined description of this menu item. |

no_cat_audit table

The no_cat_audit table, intended for internal use, maintains an audit trail of changes to the index cataloging field. The columns of the no_cat_audit table describe the attributes of the audit trail. Each row represents an update to the index cataloging field for a document class.

Contents of no_cat_audit

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_DOCCLASSNUMBER | f_docclassnumber | System-assigned document class number. |
| F_DOCCLASSNAME | f_docclassname | User-assigned name for a document class. Oracle users: If the first entry is oragen Init, the audit trail is complete. If the first entry is INX Init, the audit trail was implemented after the database was created on the date shown in the F_NO_CAT_DATE field. |
| F_NO_CAT_DATE | f_no_cat_date | Date index cataloging was turned off or on. |
| F_NOCATALOG | f_nocatalog | Status of the index cataloging field. YES = cataloging is disabled NO = cataloging is enabled |

sys_numbers table

The sys_numbers table is intended for internal use only. It merely keeps an ongoing list of the next available number to be assigned to an index column name, document class, cluster, or folder.

Contents of sys_numbers

| Column Name | Null? | Type |
|--------------|----------|--------------|
| F_SYSID | NOT null | VARCHAR2(30) |
| F_NEXTNAME | | VARCHAR2(30) |
| F_NEXTNUMBER | | NUMBER(10) |

user_index table

The columns of the user_index table describe the attributes of an index. Each row contains the attribute values of a user-defined index.

Contents of user_index

| Oracle/DB2 Column Name | MSSQLServer Column Name | Contents |
|---------------------------|----------------------------|--|
| F_COLUMNNAME | f_columnname | System-assigned identification number for a user-defined index. For example, the first index created is named A31 (Oracle) or a31 (Microsoft SQL Server), the second is named A32 (Oracle) or a32 (Microsoft SQL Server), and so on. |
| F_INDEXNAME | f_indexname | User-assigned name for an index (date, time, name, address). |
| F_DESCR | f_descr | User-assigned definition of the associated index name. The definition appears only when you are defining or modifying an index definition in Database Maintenance. |
| F_INDEXTYPE | f_indextype | The type of index: 1 = numeric 2 = string 4 = menu 8 = date |
| F_INVERTED | f_inverted | Indicates whether the index is inverted: null = not inverted 1 = inverted index With a large database, retrievals are practical only when using an inverted index (also called a retrieval key). |

Contents of user_index, Continued

| Oracle/DB2 Column Name | MSSQLServer Column Name | Contents |
|---------------------------|----------------------------|--|
| F_NOOFDUPLIC | f_noofduplic | Not used. This value is an estimate of how many duplicate inputs exist for the same index. Oracle users: Might appear in older databases: 1 = many 2 = few 3 = none |
| F_UPPERCASE | f_uppercase | Case specification for the associated string index value: null = string stored in uppercase 1 = string stored as entered in upper, lower, or mixed case) |
| F_MINIXSIZE | f_minixsize | Not used. Value of this field is always 0. |
| F_MAXIXSIZE | f_maxixsize | If the associated index type is 2 (string), this column depicts the maximum number of characters that this string can contain. For index types other than string, this value is always 0. |
| F_VALFLAG | f_valflag | This attribute is no longer used. |
| F_VALIDENT | f_valident | This attribute is no longer used. |
| F_MASK | f_mask | Template used for a numeric or date index. |
| F_NUMBERGUIDS | f_numberguids | Indicates the number of globally unique identifiers for the user index. One GUID is added automatically to the GUIDS table. min = 1, max = 10 |

Contents of user_index, Continued

| Oracle/DB2 Column Name | MSSQLServer Column Name | Contents |
|---------------------------|----------------------------|---|
| F_DMA_NAME | f_dma_name | Display name initially copied from the user index name. It can be modified but cannot be left blank. |
| F_MENU_NAME | f_menu_name | If the associated index type is 4 (menu), this column contains the name of the user-assigned menu. If it is not a 4, there will be no information in this field, because the index will not be a menu type. |

validation_tab table

The columns of the validation_tab table describe the attributes of a validation table. Each row contains the attribute values of a validation table created by the user.

Contents of validation_tab

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| F_VALTABNUMBER | f_valtabnumber | System-assigned validation table ID number. |
| F_VALTABNAME | f_valtabname | User-assigned validation table name. |
| F_DESC | f_desc | User-defined validation table description. |
| F_NUMITEMS | f_numitems | Number of items in the validation table. |

validation_tab_items table

The columns of the validation_tab_items table describe the attributes of a validation item. Each row contains the attribute values of a validation table item.

Contents of validation_tab_items

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|--|
| F_VALTABNUMBER | f_valtabnumber | System-assigned validation table ID number. |
| F_ORDINAL | f_ordinal | Sequential position of this item in the table. This is the order in which the items were created. |
| F_ITEMCODE | f_itemcode | User-assigned code for this item. |
| F_ITEMDES | f_itemdesc | User-defined description of this item, which is made available to the user as a selectable string value. |

WQS Tables

Workspace and queue descriptions are in the WQS (Oracle) or wqs (Microsoft SQL Server) database allowing the same workspace to be defined on multiple WQS servers. The WQS system tables are:

- wqs_idseed
- wqs_workspaces
- wqs_queues
- wqs_fields
- wqs_release

During an update or conversion (Oracle RDBMS only), all workspaces and queues created previously are automatically converted to include the new WQS tables.

wqs_idseed table

The wqs_idseed table stores a sequential ID number of the queues created.

Contents of wqs_idseed

| Oracle/DB2 Column Name | MSSQLServer Column Name | Contents |
|---------------------------|----------------------------|--------------------------------|
| F_IDSEED | f_idseed | ID number for multiple queues. |

wqs_workspaces table

The columns in the wqs_workspaces table store the workspaces.

Contents of wqs_workspaces

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| WS_NAME | ws_name | Workspace name. |
| WS_LEVEL | ws_level | Workspace level: 4 levels for old queues, 5 levels for current. |
| WS_TIMESTAMP | ws_timestamp | Time of creation. |
| WS_ACCESS | ws_access | Workspace security. |
| WS_DESC1 | ws_desc1 | Workspace description. |
| WS_DESC2 | ws_desc2 | Workspace description. |
| WS_DESC3 | ws_desc3 | Workspace description. |
| WS_DESC4 | ws_desc4 | Workspace description. |
| WS_ACCESS_RD | ws_access_rd | “Read” workspace security. |
| WS_ACCESS_WR | ws_access_wr | “Write” workspace security. |
| WS_ACCESS_AX | ws_access_ax | “Append/Execute” workspace security. |

wqs_queues table

The wqs_queues table columns store the queues.

Contents of wqs_queues

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| Q_WS | q_ws | Workspace name. |
| Q_NAME | q_name | Queue name. |
| Q_LEVEL | q_level | Queue level: four levels for WQM1, five levels for WQM00. |
| Q_REVISION | q_revision | Number of times queue definition has been modified. |
| Q_TIMESTAMP | q_timestamp | Creation time. |
| Q_DESCACC | q_descacc | Description security. |
| Q_CONTENTACC | q_contentacc | Contents security. |
| Q_TEXT1 | q_text1 | Queue description. |
| Q_TEXT2 | q_text2 | Queue description. |
| Q_TEXT3 | q_text3 | Queue description. |
| Q_TEXT4 | q_text4 | Queue description. |
| Q_NUMFIELDS | q_numfields | Number of user fields. |
| Q_SERVERID | q_serverid | Queue server ID number. |
| Q_TABLEID | q_tableid | Table number (nnnnn). |
| Q_DESCACC_RD | q_descacc_rd | “Read” description security. |
| Q_DESCACC_WR | q_descacc_wr | “Write” description security. |
| Q_DESCACC_AX | q_descacc_ax | “Append/Execute” description security. |
| Q_CONTENTACC_RD | q_contentacc_rd | “Read” contents security. |
| Q_CONTENTACC_WR | q_contentacc_wr | “Write” contents security. |
| Q_CONTENTACC_AX | q_contentacc_ax | “Append/Execute” contents security. |

wqs_fields table

The columns in the wqs_fields table store the queue fields.

Contents of wqs_fields

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---|
| FLD_SERVERID | fld_serverid | Field server ID number. |
| FLD_TABLEID | fld_tableid | Field table ID number. |
| FLD_ID | fld_id | Field ID number. |
| FLD_NAME | fld_name | Field name. |
| FLD_TYPE | fld_type | Field type. Numbers from 1 to 12 identify the field type: 1 number (floating point) 2 string 3 time 4 selection 5 document (document number) 6 folder (folder number) 7 integer 8 date 9 access (not used) 10 boolean |
| FLD_LENGTH | fld_length | Number of characters in the field. |
| FLD_PREC | fld_prec | Decimal precision (number of digits). |
| FLD_SCALE | fld_scale | Decimal scale (number of fractional digits). |
| FLD_UNIQUE | fld_unique | Only one occurrence of each value. |

Contents of wqs_fields

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|------------------------|
| FLD_REQUIRED | fld_required | Cannot be null. |
| FLD_RENDEZ | fld_rendez | For rendezvous queues. |
| FLD_DISPLAY | fld_display | Hidden or displayable. |

wqs_release table

The wqs_release table is defined with one column, rel_relnum, that was once used to determine whether wqs conversion tools are needed to be called during upgrade. This table only contains one row of information.

Contents of wqs_release

| Oracle/DB2 Column Name | MSSQL Server Column Name | Contents |
|---------------------------|-----------------------------|---------------------|
| REL_RELNUM | rel_relnum | WQS release number. |

WorkFlo Queue Tables

A WorkFlo queue table has six system-defined columns, followed by the user-defined columns UF000–UFnnn (Oracle) or uf000–ufnnn (Microsoft SQL Server), which describe the attributes of a WorkFlo queue. Each row contains the values associated with an item in the WorkFlo queue. See also [“Managing WorkFlo queues” on page 24](#) (Oracle users) or [page 43](#) (Microsoft SQL Server users).

Except for F_STATUS (Oracle) or f_status (Microsoft SQL Server), a WorkFlo script can set and retrieve the value in any queue field.

Contents of a WorkFlo Queue

| Oracle/DB2 Column Name | MSSQL Server Column Name | WorkFlo Parameter | Definition |
|------------------------------|-----------------------------|------------------------------|--|
| F_PRITIME | f_pritime | F_Priority and F_ETime | Priority of the item concatenated with the time it entered the queue. Because of the format, this field is intended for internal use only. |
| F_STATUS | f_status | F_Busy | Boolean field indicating if an item is busy: 0 = not busy (the default) 1 = busy |
| F_DELAY | f_delay | F_Delay | Time after which the item can be retrieved. The default is no delay with the value –2000000000 (–2.000E+09), meaning the item is eligible for retrieval without delay. |
| F_TIMEOUT | f_timeout | F_TimeOut | Time after which the item is considered too old. The default is no timeout. |
| F_USERID | f_userid | F_UserID | Three-part NCH name of a user. The default is (ANYONE). |

Contents of a WorkFlo Queue, Continued

| Oracle/DB2 Column Name | MSSQLServer Column Name | WorkFlo Parameter | Definition |
|------------------------------|----------------------------|----------------------|---|
| F_GROUPID | f_groupid | F_GroupID | Three-part NCH name of a group. The default is (ANYONE). |
| UF000-UFnnn | uf000-ufnnn | <variable> | User-defined queue field. |

Glossary

In this Glossary, terms shown in italics are glossary entries.

ageable cache

Ageable *cache* is time-limited storage on magnetic *media*. Objects remaining in ageable cache past a specified time are eligible for deletion if space is needed to store other objects. See *page cache*.

cache

Cache is the magnetic disk space used to store documents on the way to and from storage media (and can act as permanent storage when you do not use optical storage media). Portions of the cache storage are allocated to the different cache types (referred to as logical caches). See *ageable cache*, *folder notes cache*, *page cache*.

clustering

Clustering directs the FileNet system to store all documents with a common *index* value (for example, the same loan number) in a reserved space on particular *media*.

database

A database is a collection of logically related records or files. The FileNet System uses two types of databases: a third-party relational database for index data and multi-keyed databases for document addresses and work in progress. See [“RDBMS” on page 107](#).

document

Documents can be images, text, forms, mixed (combinations of types), or imported DOS files stored on the FileNet system's storage media.

document class

A document class describes the scanning, indexing, and security characteristics of a group of documents.

folder

A folder is a logical grouping of document images. A folder has a specified set of retention, disposition, and filing parameters, and a name, pathname, and ID number.

Image Services

FileNet Image Services is a set of servers and services providing a single document image *database*. The database includes a single *index database*, a single document locator database, and the collection of document images on storage media.

index

An index contains the information used for retrieving documents. All index information is stored in the index database and also on storage media in page zero of the document. Later, when you need to look at the document, the FileNet software looks in this database for index information that satisfies a retrieval query.

index database

The index database, an *RDBMS* database (Oracle or Microsoft SQL Server), contains document and folder information and can contain WorkFlo queues.

informational index

Informational index is a term used to refer to an index that is not set up as a *retrieval key*.

magnetic disk

Magnetic disk, usually an internal hard disk on your system, is where the Image Services software, *cache*, and databases are stored.

magnetic disk cache

See *cache*.

media

Media is any material on which data is stored (magnetic disk, optical disk, magnetic tape). We usually refer to optical disks as storage media.

media family

The media family defines what type of storage *media* the *document class* uses. In general, the media family controls which media surfaces will be used by the document classes that use the family.

page cache

Page *cache*, also known as retrieval cache, is a cache containing all documents being committed to or retrieved from *storage media*. In addition, documents being retrieved from media for printing are stored in page cache before being moved to print cache. Page cache is an *ageable cache*.

query

A query is a request for information or the act of requesting information from a database.

RDBMS

RDBMS is an acronym for Relational DataBase Management System. The RDBMS manages the *index database* and *WorkFlo queue database*.

retention parameters

Retention parameters specify a starting event and a number of months after that event when a document is eligible for deletion. You set up retention parameters when you create a *document class*.

retrieval

Retrieval is the act of entering a query that results in a list of documents in a query match report. Often, the process includes getting document images from the *storage library* or document indexing information from the database on the *Index server*.

retrieval key

A retrieval key is an index pertaining to certain documents to enable quick document *retrieval*.

storage library

A storage library is a storage media jukebox, a unit that has a number of slots for containing storage *media* and a robotic arm that moves the media between slots, drives, and the input/output slot.

System Monitor

An application that displays read-only reports about the state of the FileNet system. The reports are generated from data in the FileNet Management Information Base (MIB), the central *database* containing Image Services system information.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Corporation
J74/G4
555 Bailey Avenue
San Jose, CA 95141
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other

claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names might be trademarks of IBM or other companies.

U.S. Patents Disclosure

This product incorporates technology covered by one or more of the following patents: U.S. Patent Numbers: 6,094,505; 5,768,416; 5,625,465; 5,369,508; 5,258,855.

A

audit trail 91

C

case, column names 29, 47, 66

cd_id_map table 71

ce_os_dcl_map table 72

changing

 dates 33

clustering 88

CREATE VIEW statements 28, 46, 65

D

date format 33

date, converted in views 28

dclview

 DB2 65–66

 Microsoft SQL Server 46–48

doc_class_index 80

doctaba table 73

document

 access rights 74

 append/execute 75

 read 74

 write 75

 archiving 73

 CE object store id 75

 deleting 73

 disposition 74

 eligibility for deletion or archiving 74

 entry date 73

entry, IWS/CWS 76

filed in folder 86

ID number 73

pages 74

retention 28

security 29

type 28, 74

document class

 access rights 78

 append/execute 79

 read 79

 write 79

attributes 75

cataloging enabled/disabled 78

default CE Object Store ID 79

description 75

document disposition 77

document retention date 77

eligibility for deletion or archiving 77

ID number 73, 75

indexing form name 76

media family name 77

media family number 77

migration delay 79

name 75

pages per batch 76

pages per document 76

user-defined index fields 78

verification steps 76

views 27, 46, 65

WorkFlo queue 77
WorkFlo system 77

E

export log
 default CE object store id 83
export_log table 83

F

F_SW 19
Folder
 access rights
 append/execute 85
 read 85
 write 85
folder
 access rights 85
 archive and deletion dates 84
 automatic unfiled date 85
 creation date 84
 ID number 84
 name 84
 retention date 85
 table 84
folder contents
 document ID 86
 document unfiled date 86
 folder ID number 86
 order of documents in folders 86
 renumbering 86
folder_contents table 86

G

GUIDS table 87
 columnname 87

document class number 87
globally unique identifier 87

I

index_cluster table 88

L

listing
 Oracle tables 18
 queue fields 25, 44, 63
 SQL Server tables 36, 52

M

manuals, related 9
map queue column names 44
mapping user-defined queue field column
 names 25, 63
menu 89
menu table 89
menu_items table 90
multiple WorkFlo queues, ID 97

N

name, WorkFlo queue view 30, 48, 67
no_cat_audit table 91

O

output to file 26, 45, 64

Q

query language 9
queue
 description 25, 30, 45, 48, 63, 67
 fields 100

S

security

- document 29
- select statement
 - get string length 31, 49, 67
 - list tables 18
 - list views 48, 66
- spacerpt 27, 45
- SQL query language 9
- SQL*Plus, logging on 17
- string length 31, 49, 67
- sys_numbers table 92

T

- tables
 - ce_id_map 71
 - ce_os_dcl_map 72
 - doc_class_index 80
 - doctaba 73
 - document_class 75
 - export_log 83
 - folder 84
 - folder_contents 86
 - GUIDS 87
 - index_cluster 88
 - listing DB2 52
 - listing Microsoft SQL Server 36
 - listing Oracle 18
 - menu 89
 - menu_items 90
 - no_cat_audit 91
 - standard 71
 - sys_numbers 92
 - user_index 93
 - validation_tab 96
 - validation_tab_items 96
 - WorkFlo queue 102

- WQS 97
 - wqs_fields 100
 - wqs_idseed 97
 - wqs_queues 99
 - wqs_release 101
 - wqs_workspaces 98
- to_char 32
- to_date 32
- translating
 - by creating a view 27, 46, 65
 - dates 32, 68

U

- user-defined
 - date/time mask 94
 - definition 93
 - index field 75
 - index field name 93
 - index field types 93
 - index ID 93
 - inverted/not inverted 93
 - menu name 95
 - Oracle databases, older 94
 - string case 94
 - string length 94
 - validation table name 94

V

- validation items
 - code 96
 - description 96
 - position 96
 - table ID 96
- validation table
 - description 96

- ID 96
- name 96
- number of items 96
- validation_tab table 96
- validation_tab_items table 96

W

- waiting period to delete/archive 73
- WorkFlo queue table 102
 - Oracle name example 24, 62
 - SQL Server name example 43
- workspace queues
 - contents security 99
 - append/execute 99
 - read 99
 - write 99
 - creation time 99
 - description 99
 - description security 99
 - append/execute 99
 - read 99
 - write 99
 - queue definition modifications 99
 - queue level 99
 - queue name 99
 - server ID 99
 - table number 99
 - user fields 99
 - workspace name 99
- workspaces
 - access rights 98
 - append/execute 98
 - read 98
 - write 98
 - creation time 98

- description 98
- level 98
- listing 24, 43, 62
- name 98
- WQM 18
- wqm 36, 52
- WQS release number 101
- WQS tables
 - wqs_fields 100
 - wqs_idseed 97
 - wqs_queues 99
 - wqs_release 101
 - wqs_workspaces 98
- wqs_fields table 100
- wqs_idseed table 97
- wqs_queues table 99
- wqs_release table 101
- WQS_tool 24–26, 43–45, 62–64
- wqs_workspaces table 98



Product Number: 5724-R95

Printed in USA

SC19-3305-00

