Falcon/DMS

The Administrator

Operations Guide

advanced
enterprise
tsaADVET
technologies

tsaADVET

4722 Campbell's Run Road
P.O. Box 44145
Pittsburgh, PA 15205-0345
TEL: 412.787.0980
FAX: 412.787.9835

The Administrator—Operations Guide
for Falcon/DMS version 5.5
September 1999
Documentation manual written by Thomas M. Synnott

Falcon/DMS

The Administrator

Table Of Contents

Contents at a Glance

Introduction – The Introduction provides an overview of the Falcon/DMS Document Manager and Administrator.

Chapter 1 – Installation & Implementation: Chapter 1 provides a detailed workflow for the installation and implementation of Falcon/DMS on a Microsoft Windows Network. (For information on implementing Falcon/DMS on Novell and UNIX networks, see Appendix C.)

Chapter 2 – Network Configuration: Chapter 2 explains how Falcon/DMS uses the available network resources. As a general guideline, Falcon/DMS is capable of managing documents on any network operating system that supports mapped drives. On such a network operating system, Falcon/DMS supports the following options: Mapped drives (logical drive connections), Mapping-on-the fly (a non persistent mapped drive), UNC (Universal Naming Convention, and FTP.

Chapter 3 – File Formats & Applications: Chapter 3 introduces File Formats and Applications. The File Format attribute assigned to each document record as it is added to the database indicates which application (or in certain cases, which applications) is used to open that document. However, the application association is only the most obvious function of the File Format attribute: this chapter also provides detailed explanations File Association, File Format Mapping, and non-electronic document control.

Chapter 4 – MassAdd & the Master 'Project' Table: Chapter 4 explains how you use the MassAdd utility to first, create the Master 'Project' Table, and second, to 'bulk load' your legacy data (that is, to create database records for existing documents). The chapter concludes with a discussion of how you can maintain your data structure from the Master 'Project' Table environment.

Chapter 5 – General Configuration: Chapter 5 provides a detailed explanation of basic configuration issues that includes, among others, System Wide Toggles, the environment that controls such critical parameters as filename and directory name lengths, and System Labels & Templates, used to customize the labeling schemes on the administrator and user interfaces,

Chapter 6 – Security: Chapter 6 introduces the issue of security. Falcon/DMS divides security into three categories: Operational Privileges, as defined through Account and Access Control, Document Status Access, which allows you to set one of three access modes (modify, view access, or no access) to specific files or groups of files, and Project Security, used to establish project-level security schemes that controls access to project directories or sub-directories within projects. The First two categories, Accounts/Access Control and Document Status Access are presented in Chapter 6. Project Security is discussed is Appendix D.

Chapter 7 – Multiple Environments: Chapter 7 covers Multiple Environments. This feature allows different groups, departments, or individuals to work with a set of Falcon/DMS parameters and a user interface configuration specific to their unique requirements. Chapter 7 explains the procedures you will follow to design and implement environments, and how to assign groups and individual users to one or more environments using the Environment Security tools.

Chapter 8 – Database Maintenance: Chapter 8 focuses on the tools available to maintain and modify the database: the Database Maintenance dialog box, the environment that gives the administrator direct access to the most sensitive fields in the database, the database export and import tools, and database verification. The Database Maintenance dialog box gives the administrator direct access to the most sensitive fields in the database. The import command allows you import valuable descriptive data, if it exists. The export command allows you to export specific fields from the Falcon/DMS database to a delimited ASCII file. Finally, the Database Verification command enables you to easily identify database discrepancies by listing database records for which matching, server-based, files are not found and listing server-based files for which database records do not exist. (Because helps to identify files yet to be added, the process of verifying server-based files against the database is particularly useful during the MassAdd step of a Falcon/DMS installation.)

Chapter 9 – Database Keywords: Chapter 9 introduces the concept of database keywords, a database attribute that provides valuable descriptive information about a document and, if used properly, can significantly enhance the use of Falcon/DMS's database search tools. Falcon/DMS provides two categories of keywords: System-level and File Format-based keywords. This chapter discusses the design, implementation, and use of database keyword schemes for both keyword categories.

Chapter 10 – Location & Archive Commands: Chapter 10 explains the Location Checkout, Location Check In, Archive, and Unarchive commands. The Location Checkout command is used to deliver documents to a destination outside the environment controlled by the Falcon/DMS while maintaining database records identifying the location to which the documents were sent. The Archive command is used to select files and move them to a location from which they are archived using a method of the administrator's choosing. You can choose to keep the database records of archived documents on-line; in addition to the standard descriptive information, those records will also identify the archive volume and archive data.

Chapter 11 – MicroStation & AutoCAD: Chapter 11 details the configuration issues unique to MicroStation and AutoCAD. Included in the discussion is an explanation of how Falcon/DMS controls MicroStation reference files and AutoCAD Xrefs.

Chapter 12 – Revision Control: Chapter 12 explains how to implement a revision control scheme. Revision Control is designed to track revisions in one of two modes—a History mode and an As–Built/Work in Progress mode.

Chapter 13 – Reports: Chapter 13 covers both the Main Database Reports utility, the standard report tool, and Session Accounting, an optional module that performs a broad range of auditing functions.

Appendix A – Implementing an SQL database: Provides an overview of the installation and configuration of Microsoft SQL Server and Oracle.

Appendix B – Upgrading from Falcon/DMS v4x to v5: Outlines the procedure used to upgrade Falcon/DMS version 4 to version 5.

Appendix C – Configuring Falcon/DMS for Unix and Novell: Provides an explanation of how Falcon/DMS is configured for Unix and Novell network operating systems.

Appendix D – Project Security: Explains how to configuring and implement Project Security.

Appendix E – Falcon/DMS Server Permissions: Defines the explicit file per-missions for the Falcon/DMS server components on the application server.

Appendix F – ASCII Import: Lists all the database fields available for the Export and Import ASCII commands.

CONTENTS

Falcon/DMS

The Administrator

Introduction

Because most organizations now depend on a computer network to conduct their day-to-day business, the issue of document management has become central to the task of information management. As advances in network technology make the fast and reliable transfer of data available to more and more people, the problems that accompany the control of this information grow. With our latest version of Falcon/DMS, we offer organizations a comprehensive and flexible solution to these problems that remains easy to implement, customize and use.

Introduced in 1989, the Falcon/DMS Document Management System is the acknowledged leader in both design file and general document management in the MicroStation CADD community, providing comprehensive control over both documents and plotting for some of the world's largest CADD users. With the introduction of its AutoCAD version in 1993, Falcon/DMS was established as the complete solution for document control in the two broadest segments of the engineering and architectural CADD markets. tsaADVET continues to focus on this key market, even as the capabilities of Falcon/DMS expand beyond that of the complex tasks of CADD file and plot management.

Today, however, the Falcon/DMS technology has evolved such that we now offer a complete solution for most office automation efforts. An example of this 'complete solution' design is the Windows Document Manager, which now provides comprehensive file management, SQL search, and indexing control for *any* document, *any*where, on *any* network. The scanning and print/plot modules, and DMSView, a view-index-redline application, are seamlessly integrated with the Document Manager and, when implemented, make it possible to easily manage both paper and electronic documents from the same environment. Falcon/DMS interacts directly with the Oce 9400 and 9800 network printing solutions, CADNET's Plot Station, and Intergraph's IPLOT and APLOT engineering plotting systems as well as the ANA*tech* Eagle scanner.

Falcon/DMS v5.5

While it now meets an expanded range of document management needs, Falcon/DMS remains seamlessly integrated with the MicroStation and AutoCAD applications, and is able to support both file formats in the same, or separate, databases. Falcon/DMS also continues to provide the best-in-industry tools to help designers take full advantage of MicroStation's reference file and AutoCAD's Xref technologies. Additionally, the powerful File Refresh commands insure that all MicroStation users have access to the most current reference file information

From an administrator's perspective, the already powerful and flexible feature set is even more robust—Falcon/DMS now supports file format-specific database keywords, an SQL-driven reporting engine, revision control, long filenames and directory names, UNC and FTP network configurations, Multiple Environments and Project Security. Our Access Control and Document Status Access tools continue to provide an unrivaled scope of control, ranging from how many files a user can check out to designating a set of administrative tools for supervisory personnel.

Major enhancements have also been made to MassAdd, the program designed to create database records for existing documents. MassAdd now scans designated server drives and automatically builds the Master Project Table, the list that contains the project directories under which you store data files. After building this table, MassAdd scans the listed directories and their subdirectories and adds files based on extensions that you designate.

The 'core' Falcon/DMS product is comprised of four environments:

❏ Document Manager—the user's interface.
❏ Administrator—for configuring and administering Falcon/DMS.
❏ MassAdd—easily and efficiently 'bulk loads' the database.
❏ Reports—reports module powered by a runtime version of Crystal Reports.

❍ Session Accounting—an optional auditing module.

Let's examine the main features of each, starting with the Windows Document Manager.

The Document Manager provides an extensive set of document and database management tools—featured on the following pages are several of the Document Manager's user and administrator functions:

The Document Manager

Application Independent

The Document Manager is application independent and capable of managing documents for any Windows-based application. When a file is identified and the Open command selected, that document's database record identifies the application to launch based on the record's file format. File formats and application properties are defined from the Falcon/DMS Administrator.

Falcon/DMS can automatically launch MicroStation when a user selects a MicroStation design file, or it can be configured to open a variety of MicroStation 'flavors', to include different workspaces and/or different engineering application configurations.

A comprehensive command set

The Document Manager provides users with rapid access to an full complement of file management operations. Access to all commands can be controlled as 'operational privileges' that are tailored on a per group or per user basis.

Robust searching & indexing tools

The Document Manager provides an SQL search tool able to sort on any single record field or combination of fields. Additionally, the Database Search command supports the standard SQL comparison operators for date and number fields.

For administrators, the Document Manager provides a Mass Update function that permits you edit database records for multiple documents. Mass Update can be used very effectively in conjunction with the Database Search command.

Six system & twenty file format keyword categories

A 'keyword' is a database attribute that adds to the descriptive value of a document record and significantly enhances your ability to refine the criteria used to search a database.

Falcon/DMS support six system level and twenty file format keyword categories and an unlimited number of keywords within each category. (It is important to distinguish between a keyword category and the keywords listed under each category—while there is a fixed number of keyword categories, there is no limit to the number of keywords that you can define for each category.) By requiring the entry of keywords for a variety of document management operations, keywords can be used in an ongoing effort to easily populate your database with valuable descriptive data. Additionally, you can automatically validate keyword input by limiting the selection of valid keywords to those found in administrator-defined lists. Finally, for required entries, Falcon/DMS identifies which database fields are required for a given operation, making it easy for users to comply with your standards.

The File Format Keyword feature enables you to associate a unique set of keyword categories you to each file format. For example, you could define different sets of twenty keyword categories for each of your CADD file formats, a separate set for your word processing file format, and a separate set for your spreadsheet file format.

Easily switch between a database & a directory view of your data

Falcon/DMS supports both a database view and a directory view of your production documents. This enables the Document Manager, the user interface, to list only those documents for which Falcon/DMS has database records, or to list the actual contents of server-based directories, including files not registered in the database. The flexibility this feature adds to the administration of Falcon/DMS is significant. For users, you are able to restrict the list of server-based documents to those with database records. Administrators, however, can see the physical directories and easily create database records for files located on the server. For example, documents from an outside consultant are placed directly into their server-based home directory. Though users cannot see the files until database records are created, the files are immediately visible to the administrator. Using the Document Manager, the administrator can then easily create the records as required.

Multiple Environments

The Multiple Environments feature allows different groups or departments (or for that matter, individuals) to work with a configuration and a user interface specific to their unique requirements.

File Association

In addition to comprehensive support of MicroStation's and AutoCAD's reference file technologies, Falcon/DMS also provides an application-independent file grouping function call File Association. This feature is used to define a many-to-one relationship between an unlimited number of 'child' files and a single 'parent' document. Associations to a 'parent' document are listed when you open a document for which associations

✉ NOTE
You also have the option of an internal MDL interface for MicroStation. This is supported for DOS, Windows, and Windows NT. An internal ADS interface is available for AutoCAD r12 on DOS. AutoCAD r13 and r14 link directly to the standard Falcon/DMS Document Manager. The File>New and File>Open commands for both MicroStation and all versions of Auto-CAD can be redirected to Falcon/DMS.

have been defined. File Association allows you to choose which files are to be delivered to the local workstation, and in the case of MicroStation design files and AutoCAD drawing files, which are to be checked out for modification and which are to be copied out for view. An especially powerful feature of File Association is the ability to create database records and associations for documents created on the local hard disk during a design or work session.

Revision Control

Revision Control is designed to track revisions in one of two modes—a History mode or an As–Built/Work-in-Progress mode. The key difference between modes is in how each identifies the current revision.

Revision Control schemes can vary between file formats and can be enabled for some and not for others. Revision Control can be configured to create a new revision at file checkout or file check in, either automatically or at a prompt. Revision Control can also be configured to allow you to create a revision at your discretion.

Intelligent Data Link—Bi-directional database linkage of metadata and MicroStation's graphical data

IDL, or Intelligent Data Link, is a database feature designed to automatically define the contents of a design file's title block. Integrated with the Create Design File command, IDL uses information entered in the General Info, Directory & Keywords, and Title Block 'tabs' found on both the Create Design File dialog box and the Document Manager.

The Intelligent Data Link is based on an IDL template file that is automatically linked to a MicroStation seed file of the same name. Using MicroStation's text node technology, the IDL template file maps database fields to predefined text nodes in the linked seed file.

IDL is bidirectional—the text displayed on targeted text nodes in a MicroStation design file can be modified from the document's database record, or the database record can be modified by editing the text nodes with MicroStation's Edit Text command.

The IDL template file is configured from the Administrator's Intelligent Data Link Configuration dialog box.

For the Administrator—Location & Archive Tools

The Document Manager provides a set of administrator tools that allow you to manage documents that are either sent to a location outside your organization or taken off-line. For example a client may need to approve a complete set of fabrication and specification documents. For this purpose you will use the Location Checkout command to maintain a read-only version of the document while your client completes his review. The Location Check in command enables you to return the files to the database and automatically restore the documents's access mode to read_write.

The Archive command allows you to select files and move them to a location from which they are archived using a method of your choosing. The date of the archive and the directory to which the documents were moved are captured by Falcon/DMS and written to their database records.

The Administrator

From the Administrator you will define and maintain the database, configure the network options and system-wide toggles, define or edit file formats and application properties, and create user accounts and setup security.

The Administrator tools are divided into four general categories:

❏ Database
❏ Tables
❏ Security
❏ Configuration

Database:

Database maintenance

The Administrator's Database Maintenance dialog box provides access to every field in each document's database record, as well as the ability to view a file's recent history. If necessary, you are able to edit all database record fields for any document from this environment.

Importing & export database records

The Administrator's Database Maintenance environment provides access to every field in each document's database record, as well as the ability to view a file's recent history. If necessary, you are able to manually edit all database record fields from this dialog box.

Verification

If the files that Falcon/DMS will control have a wide variety of extensions, keeping track of what has been added to the database can be a challenge. The Database Verification utility provides an easy way to determine which existing files do not yet have database records. This utility scans the file repositories defined in Falcon/DMS and lists all files without database records. Database Verification can also be configured to identify database records not pointing to an actual file.

Tables:

Master 'Project' Table

The Master 'Project' Table is used to add individual projects to your Falcon/DMS configuration. The MassAdd program, as you will see, has multiple functions, one of which is to 'bulk load' the project table. For adding and managing individual entries you will use the Master 'Project' Table dialog box.

When the Falcon/DMS network configuration is set to either mapped drives or UNC, the Master 'Project' Table dialog box can also be used to create the actual server-based directories and their subdirectories for new projects in your data structure.

System labels and templates

Falcon/DMS permits you to organize your documents in a directory structure that can be from one to six levels deep and to define your own labels for each directory level. If your business is a manufacturing concern you may choose to organize your files within 'product' categories, while an engineering company might organize its files by 'project' or 'client'— Falcon/DMS doesn't restrict you.

Security: The Administrator provides two categories of security features: 'operational privileges based on Access Control, and 'file privileges' based on Document Status Access.

Accounts & Access Control

Access Control allows you to specify such parameters as the number of files a user can retrieve at one time, the ability to create or delete documents, or the level of access a supervisor has to a projects database records, to name just a few. Access Control permits you to tailor operational privileges based on job description, from the basic file open and check in capabilities of a junior level user to the assistant administrator responsibilities of a department manager.

Document Status Access

Access to documents can be controlled on the basis of Document Status, a database attribute that, under certain conditions, 'enhances' explicit file permissions granted by the network operating system. For example, if a set of documents have been forwarded to an engineer for review, access by other users to this set of files should be limited to view only. Thus, when a designer who normally has sufficient privileges to modify documents opens a design file that has been 'Submitted For Review', that designer, is unable to alter the document without receiving authorization from the engineer.

Configuration: Most of the basic and advanced configuration issues are addressed from the options located on the Administrator's Configure menu. Some of the most important categories are:

Environments

The Multiple Environments tools are found in two categories: Configuration and Security.

An Environment Configuration is comprised of settings defined from the Administrator for the labeling schemes and option lists that appear on the Document Manager. Simply put, Multiple Environments offers multiple ways of organizing these configuration files.

An environment definition can use existing configuration files and tables (global definitions) for certain controls and build unique configuration files and tables for others. Also, Falcon/DMS can use any of the existing global definitions as a template for a new definition, making the Multiple Environments feature easy to configure.

Environment Security gives administrators the ability to control access to the different configurations defined within Multiple Environments. With Environment Security you can assign these specific configurations to different groups within the organization. Also, Environment Security can be used to assign specific configurations to individuals, regardless of his or her group membership. (This is particularly useful when Multiple Environments is used to define different Master 'Project' Table configurations and/or keyword schemes.)

File Formats & Applications—a Simple Association

Falcon/DMS supports virtually all Windows application and is DDE compliant. This means that your organization can use the Falcon/DMS Windows Document Manager to manage any Windows document and automatically launch the application necessary to open that document. The File Formats & Applications dialog box is used to determine how the Windows Document Manager handles this task.

System Wide Toggles

The System Wide Toggles dialog box is used to establish a number of important parameters that affect the most fundamental ways in which Falcon/DMS operates. Included among these controls are:

Support for long filenames and directory names: Falcon/DMS supports a maximum filename length of eighty (80) characters and a maximum directory name length of forty (40) characters. It is important to understand that each component, or subdirectory, of the full directory path specification can contain up to forty characters.

Instant recovery for deleted files: Falcon/DMS is usually configured to move files to a secure directory on the network when the Document Manager is used to delete documents. Rather than deleting the actual files, Falcon/DMS gives that responsibility to the administrator, who removes the files after sufficient time has elapsed to insure that the documents are, indeed, no longer needed.

Keyword validation: When adherence to standards is required for assigning descriptive database attributes, users able to assign only those keyword appearing in predefined lists. If a user chooses to key in a keyword, that item must appear in the associated list box.

Location Check in and Location Checkout parameters: When the Location commands are made available to users not in the administration group, you will often want to control what location options are available. The System Wide Toggles dialog box allows you to lock certain fields and options on the Location form to insure that use of the Location commands comply with your standards.

Enabling of Hook Functions and File Association: The Hook Functions control and the File Association control are located in System Wide Toggles.

Network Configuration

Falcon/DMS supports mapped drives, UNC (Universal Naming Convention) mapping and FTP, and utilizes existing network resources. Falcon/DMS requires no network tuning beyond that which you perform to optimize network operating system performance. Please note that the Network Configuration environment defines the manner in which Falcon/DMS interfaces with your network—in no way does the Falcon/DMS Network Configuration alter your you network or its settings.

Session Accounting Toggles

The Session Accounting Toggles dialog box is used to select which document management operations are to be tracked and what type of sessions are captured.

Hook Functions

The Hook Functions dialog box is used to identify and configure the 'hooks' that custom applications use to communicate with Falcon/DMS. A complete set of 'pre' and 'post' functions (e.g. pre_create, post_create, pre_open, post_open, etc.) are provided for the MicroStation (MDL), AutoCAD (ADS) and the Windows versions of the Document Manager.

MassAdd

Legacy Data Loaded With Ease

MassAdd is designed to create database records for legacy documents. MassAdd also scans designated server drives and automatically builds the Master Project Table, the list that contains the project directories on the file repository.

Typically, the first step in the MassAdd procedure is to build the Master 'Project' Table and the second step is to bulk load the database. By 'bulk load' we mean that, in a single operation you create database records for files located in one, or several, or all 'project' directories. As MassAdd creates each record, it associates that file with the appropriate file format, which Falcon/DMS uses to identify the application needed to read that document.

Though launched from the Administrator, the Database Verification utility has become an essential part of the MassAdd process, especially for database bulk loads of legacy data comprised of files with many different file extensions. For example, while MicroStation files default to a '.dgn' extension, MicroStation will open any design file regardless of its extension. For this reason many organizations have used the design file's extension as a filename suffix rather than accept the default. Because Database Verification scans the project directories and identifies files that do not have matching database records, it is effective in identifying files that have not been loaded through MassAdd.

Reports

Session Accounting & Main Database

Session Accounting is an add-on module used to audit any combination of Falcon/DMS operations and is capable of tracking session times for MicroStation and AutoCAD. This time-in-file feature is configurable so that auditing is suspended after an administrator-defined period of inactivity elapses; auditing resumes when keyboard input or cursor movement is detected. This feature is used by organizations for both billing purposes and productivity studies.

The Main Database Reports is powered by a run-time version of Crystal Reports, a standard report application capable of reading all the databases supported by Falcon/DMS. The report engine is fully accessible with a licensed version of Crystal Reports for the purpose of generating fully customized reports. However, because the database search tools are fully integrated with the reporting environment, you are able to search on any combination of fields from the Main Database Reports utility delivered with Falcon/DMS.

Falcon/DMS

The Administrator

Chapter 1:
Installation &
Implementation
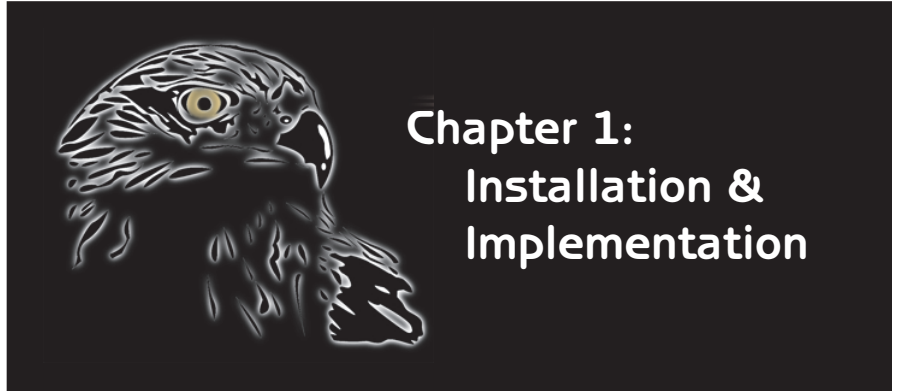
The Falcon/DMS server and client installation is a four phase process.

In Phase I you will prepare your application server and install the Falcon/DMS server components. Because it is necessary to create and share a server directory, your logon account requires administrator-level privileges.

The explanation of the installation workflow, starting on the following page, assumes that your application server will not be used to administer Falcon/DMS. While there is nothing to preclude administering Falcon/DMS from the application server, for the purpose of explaining the installation process, a different computer is designated as the administrator's workstation.

In Phase II you will define the necessary environment variables and install the appropriate components on the administrator's workstation.

In Phase III you will use the Administrator application to establish the basic operating environment necessary for Falcon/DMS to communicate with the database and the data repository. In Phase III you will also define the Master 'Project' Table and create database records for legacy, or existing, data.

Finally, in Phase IV you will define the necessary environment variables and install the user components on a client workstation.

The installation workflow assumes that a database has been installed and configured on the database server, and any client components required by that database are installed and configured on the Falcon/DMS administrator's workstation as well as on the users' workstations. For Windows-based operating systems, the workflow will also discuss the basic ODBC configuration issues.

The databases for which Falcon is certified are Oracle (v7.3.3 and 8x), Microsoft SQL Server v6.5 and 7x), and Sybase (v10). (Please refer to Appendix A for basic configuration information on Oracle and Microsoft's SQL Server)

For users upgrading from Falcon/DMS version 4 to version 5, please refer to Appendix B.

Installation Workflow

Phase I

[Perform the following steps on the application server.]

1. Create the Falcon/DMS directory.
2. Share the Falcon/DMS directory.
3. From the product CD, run SETUP.EXE.

Phase II

[After logging on as a local administrator, perform the following steps on the administrator's workstation.]

1. Define the required environment variables.
2. Map a network drive to the application server.

(Though this mapped drive is optional, the convenience of direct access to the application server simplifies the installation and maintenance of Falcon/DMS.)

3. From the mapped network drive, run SETUPWIN.EXE to install the components necessary to administer Falcon/DMS.

Phase III

[Perform Step 1 from the ODBC Administrator]

1. Define a System DSN (Data Source Name).

[Perform Steps 2-5 from the Falcon/DMS Administrator.]

2. Set the Data Source and the database login account and password.
3. Define the database's Primary Index Key and create the Falcon/DMS tables.
4. In the Network Configuration environment, set the configuration to UNC and define the server and resource tables:
5. Define the File Format table.

[Perform Steps 6 & 7 from the Falcon/DMS MassAdd utility.]

6. Define the Master 'Project' Table.
7. Bulk load legacy documents (existing, server-based, electronic files) into the database.

Phase IV

[After logging on as a local administrator, perform the following steps on each client workstation.]

[Perform Step 1 from the ODBC Administrator and Steps 2-4 from the operating system.]

1. Define a System DSN (Data Source Name).
2. Define the required environment variables.
3. Map a network drive to the application server.

Again, though this mapped drive is optional, it is convenient for the maintenance of Falcon/DMS and is available only from the local administrator's account.)

4. From the mapped network drive on the application server, run SETUPWIN.EXE to install the Falcon/DMS user components.

Phase I

In Phase I you will prepare your application server and install the Falcon/DMS server components.

The operating system for the application server can be Windows NT Server, Novell, or UNIX. The basic preparations are similar regardless of the operating system; the exact procedures may differ between operating systems but the principles are the same.

The following workflow description is for an application server running Windows NT Server and client workstations running Windows NT Workstation and Windows 95/98. (Please refer to Appendix C for Novell and UNIX installations.)

Step 1

On the application server, create a directory for the Falcon/DMS server software.

For example: [C]:\dms_serv, where [C] is the target drive and 'dms_serv' is the target directory into which the setup program installs the server software.

Step 2

Share the dms_serv directory. This allows client workstations access to configuration files through UNC.

Step 3

Run the Setup program to install the Falcon/DMS server and client components.

1. Select Start > Run. On the Run dialog box, enter [D]:setup, where [D] is your CD-ROM drive, and press OK.

2. From the Falcon/DMS Installation Options form, select the New Installation... option.
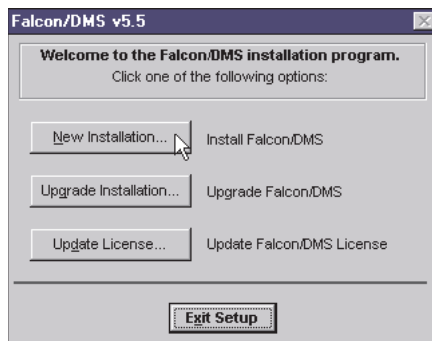
Additional options:

Upgrade Installation... permits you to upgrade existing installations without altering current configuration files.

Update License... permits you to increase the number of client seat licenses and/or install the Session Accounting module.

3. In the 'Install to' field, enter the directory created in Step 1 (the default value is M:); for example: C:\dms_serv.

Press OK.

4. Enter your Falcon/DMS license number. Press OK.

<div class="note">

✉ NOTE

Though this workflow description assumes the opposite, it is not unusual for the application server to double as the Falcon/DMS administrator console, at least until the system is configured and the database seeded. In such cases, both the server components and the client interface will be delivered to the application server.

The directory structure below shows a typical installation path for a Windows NT implementation in which the application server is also used to administer Falcon/DMS:

[C]:\falcon5\dms_serv

The advantage of this directory structure lies in the fact that it consolidates the client and server components while keeping the two separate.

To illustrate this, let's break it down into its component parts:

❏ [C] is the target drive.
❏ falcon5–location of client software.
❏ dms_serv–location of server software.

</div>

5. The Setup Option form opens. From this form, choose the components that you wish to deliver to your application server. (After this delivery is complete, you will use the Client Setup form to install the user interfaces. The Client Setup is explained in Phase II.)

Setup options:

Database Type—Press the Microsoft SQL, Oracle, Sybase, or Btree radio buttons:

When Microsoft SQL, Oracle, or Sybase is selected, you are reminded that the relational database must be installed and configured before Falcon/DMS can create the necessary tables.

When the Btree option is selected, Falcon/DMS installs the non-relational database engine delivered with the product. Be advised that, while Falcon/DMS continues to provide support for the Btree database, many features available in an SQL database implementation are not supported for Btree.

Falcon/DMS Core—Select the appropriate check boxes:

Falcon/DMS Core Product—installs the Administrator and Windows Document Manager tools.

Session Accounting Module—installs the add-on module that provides system auditing, the ability to track and report on a wide variety document management activities (see Chapter 14).

Report Module—installs the standard reports module. This includes a run-time version of Crystal Reports and a set of standard report forms (see Chapter 14).

Online Documentation—installs the Adobe PDF version of the documentation to the application server. Included on the product CD is the Version 4 of the Adobe Acrobat Reader.

MicroStation Modules—These interfaces are MDL applications completely integrated with the MicroStation environment. Select the appropriate check boxes:

Windows 95/98/NT; Windows 3x; DOS; CLIX; Sun OS; Sun Solaris; Hewlett Packard.

AutoCAD Modules—Select the appropriate check boxes:

Windows 95/98/NT; Windows 3x; DOS; Sun OS; Sun Solaris

From the Options form, select the required installation options and press Continue.

6. The Setup message box opens, press Cancel to complete the server installation.

   When you press OK on the message box, SETUPWIN.EXE, the client installation application, launches. Because, for the purpose of this explanation, we assume that the Falcon/DMS Administrator application will not be installed on the application server, you will run SETUPWIN.EXE in Step 3 of Phase II.

Phase II

In Phase II you will prepare the administrator's workstation and install the Falcon/DMS Administrator, the Main Database and Session Accounting Reports modules, and the Windows Document Manager.

The operating system on the Falcon/DMS administrator's workstation must be Microsoft Windows NT (Server or Workstation), Windows 98, or Windows 95.

Step 1

On the Falcon/DMS administrator's workstation, define the required environment variables.

A minimum of two environment variables is required for Windows NT while a minimum of four is required for Windows 95 and 98 clients. The 'dcspcnod' and 'username' variables required for Windows 95 and 98 are the equivalent of the system–defined Windows NT variables, named 'workstation' and 'username'.

Modify the path variable only when the MicroStation component is installed locally.

Windows NT—a minimum of two variables is required:

1. From the Control Panel, select System > Environment

2. Define the following as *system* variables:

VARIABLE	EXAMPLE VALUE	DESCRIPTION
falcon_dir	[C]:\falcon5	This variable points to the local Falcon/DMS directory. The SETUPWIN.EXE program creates this local directory and several subdirectories, and delivers a small set of configuration files to this location.
falcon_sdir	\\server\dms_serv	This variable points to the server components on the application server. The example value shows a UNC path to the dms_serv directory on your application server. You may also use a mapped network drive as the falcon_sdir value
path	[C]:\falcon5\mdlapps	[Optional] The location of the local MicroStation components.

Windows 98/95—a minimum of four variables is required:

1. Open autoexec.bat.

2. Add the following variables [syntax: set variable = value]:

VARIABLE	EXAMPLE VALUE	DESCRIPTION
falcon_dir	[C]:\falcon5	This variable points to the local Falcon/DMS directory. The SETUPWIN.EXE program creates this local directory and several subdirectories, and delivers a small set of configuration files to this location.
falcon_sdir	\\server\dms_serv	This variable points to the server components on the application server. The example value shows a UNC path to the dms_serv directory on your application server. You may also use a mapped network drive as the falcon_sdir value.
username	your logon account	Required to audit transactions on the current workstation.
dcspcnod	computer name	Required to audit transactions on the current workstation.
path	[C]:\falcon5\mdlapps	[Optional] The location of local MDL components.

Step 2—optional

Define a persistent mapped drive to \\server\dms_serv.

The SETUPWIN.EXE program, explained in Step 3, can be launched from the Start > Run command with a UNC path. However, it is likely that you, as the administrator, will require direct access to the Falcon/DMS directory on the application server. For this reason we recommend that you establish a persistent mapped drive to dms_serv, on both the administrator's workstation and the user's workstations; the users will not be able to see this mapped drive. Because environment variables are set at the system level, you will need to log on to client workstations as a local administrator. As such, the administrator's account can be the only account with a drive mapped to dms_serv.

Step 3

1. If you chose not to map a drive to your application server, select Start > Run. In the Open field, key in the following and press OK:

   \\server\dms_serv\install\setupwin

   where [server] is your application server and [dms_serv] is the shared directory containing the Falcon/DMS server software.

   If you have a mapped drive to dms_serv on your application server, open the Windows NT Explorer and navigate to the [M]:\install directory, where [M] is mapped to \\server\dms_serv.

   Double-click on SETUPWIN.EXE.

2. The Setup Options form opens; choose the components to deliver to the administrator's workstation.

Setup options:

Administrator—Select all the check boxes:

❏ Administrator: the administrator's interface.
❏ Mass Add: used to build the master 'project' table and bulk load the database.
❏ Main Database Reports: the standard reports module.
❏ Session Accounting Reports: provides system auditing (an add–on module).
❏ Run Application:
   Local: Copies the selected applications to your local hard disk
   Server: Runs the selected applications from the server.

Document Manager (Windows)—Select the Document Manager check box:

The Windows Document Manager is necessary to properly administer your Falcon/DMS installation. In addition to standard file management commands, the Document Manager provides the administrator with the ability to easily create document records, process files for Location Checkout and Check in, and Archive and Unarchive. Access to the Location and Archive commands is typically limited to administrators (see Chapter 13).

❏ Run Application:
   Local: Copies the Document Manager to your local hard disk.
   Server: Runs the Document Manager from the server.

AutoCAD (if required): when selected, this option modifies AutoCAD's main menu file (acad.mnu) and main lisp file (acad13/14.lsp), placing a pull–down menu on the AutoCAD menu bar that links to the Windows Document Manager. AutoCAD's File > Open and File > New commands are redirected to the Windows Document Manager when the main lisp file is modified.

❏ AutoCAD Path: press Browse to define the location of AutoCAD.
❏ Press the R13 or R14 radio button to identify the version of AutoCAD installed on the administrator's workstation.

Document Manager (MDL)[if required]—Select the Document Manager check box:

Note: This option installs an MDL version of the Document Manager that is fully integrated with MicroStation's MDL interface.

❏ MicroStation Path—press Browse to define the location of MicroStation.
❏ Run Application:
 Local–copies the Document Manager to your local hard disk.
 Server–runs the Document Manager from the server.

Phase III

In Phase III you will establish the basic operating environment Falcon/DMS needs to communicate with the database and the data repository. You will also define the Master 'Project' Table and create database records. Let's consider each piece of this operating environment separately.

ODBC

Falcon/DMS uses ODBC to communicate with the relational database. To configure an ODBC data source, the Falcon/DMS administrator must enter, along with other information, the login ID and password for the portion of the database reserved for the Falcon/DMS tables. This account is usually defined by the database administrator at the time the Falcon/DMS portion of the database is built. The database login ID and password will also be used by the Falcon/DMS application that was installed in Phase II. When Falcon/DMS is installed, a default data source, login account, and password are defined in the Administrator's Database Account form. It is likely that these default values, certainly for the login ID and password, will not match the values assigned to the ODBC data source and to the database login account and password.

In Step 1 we discuss what must done to insure that the ODBC data source can communicate with the Falcon/DMS portion of the database. In Step 2 we explain what must be done to insure that the Falcon/DMS Administrator can communicate with the database through the ODBC data source.

The Falcon/DMS Database Tables

You will use the Falcon/DMS Administrator to automatically create the necessary database tables. However, before creating the tables you must address the issue of filename uniqueness.

Every database table has a primary key. A primary key is a column or set of columns in one table that uniquely identifies each row, or record, in that table. Every record in a table has a value in the primary key that is different from that of every other record in that table. In Falcon/DMS, the most common primary key is the column containing the filename. However, under certain conditions the primary key will include, in addition to the filename column, one or more columns containing the individual subdirectories that comprise the filename's directory path.

As it relates to filename, the primary key reflects perhaps the most basic condition affecting how you organize your data: Will Falcon/DMS permit the use of duplicate filenames?

We use the term 'filename uniqueness' to refer to the level of restrictions placed on the use of duplicate filenames. Falcon/DMS allows up to seven levels of filename uniqueness.

When Falcon/DMS creates the database tables it assigns a primary key (what we refer to as the 'Primary Index Key'). This Primary Index Key indicates the level of filename uniqueness and will directly affect how Falcon/DMS creates database records for existing

data. The question you must ask is: "Do I want each filename to be used only once within my database, or, do I want to be able to use that filename more than once?"

If you decide not to allow duplicate filenames, and instead require unique filenames, your level of filename uniqueness is 1 and your database's Primary Index Key must be set to 'Filename'. Most organizations using Falcon/DMS stipulate unique filenames.

If you decide to allow a specific filename to occur in every directory on your data repository, your level of filename uniqueness is 7 and your database's Primary Index Key must be set to 'DirLevel1;...;DirLevel6;Filename'. Very few organizations adopt this scheme. And because the Primary Index Key has a direct effect on how MicroStation's reference file attachments are resolved, certainly fewer in which Falcon/DMS is used to manage MicroStation documents. (The issue of resolving reference file attachments is discussed Chapter 12.)

To explain how the different levels of filename uniqueness effect the use of duplicate filenames, lets begin by examining the directory structure supported by Falcon/DMS.

Your directory structure can have as few as one level and as many as six levels—for the sake of explanation we will refer to this top level as the 'project' directory (you can of course have as many project level directories as you require):

\\file_server\share\Project100

```
├─ SubDirectory Level 1
├─ SubDirectory Level 2
├─ SubDirectory Level 3
├─ SubDirectory Level 4
└─ SubDirectory Level 5
```

It is important to recognize that there is no relationship between the number of subdirectory levels you permit under a project directory and the level of filename uniqueness. Your level of filename uniqueness could be 1, or 3, or 7, and each project's directory structure could still drill down to a depth of six levels; the depth of a project's path has no direct bearing on your decision regarding a Primary Index Key.

When you assign a Primary Index Key based on a filename uniqueness level of 2, 3, 4, 5, or 6, you are actually telling Falcon/DMS at what subdirectory level under each project are duplicate filenames permitted. (Recall that a filename uniqueness level of 1 stipulates unique filenames and a level of 7 permits the same filename to be used in any directory.)

To explain the difference between the various levels of filename uniqueness, let's begin by examining a database with a Primary Index Key of 'DirLevel1;Filename', which has a filename uniqueness level of 2. Again, we will refer to directory level one (DirLevel1) as the 'project' directory.

As you can see in the scheme above, the project directory (Project100) is located under the shared directory on the data repository. Let's say you have one hundred projects; the number of subdirectories within each project varies—some project don't have any subdirectories while others, like Project 100, have the maximum of five.

With a filename uniqueness level of 2, the same filename (e.g. ABC123.DGN) can appear at any subdirectory level within each project, but only once within that project. That means that the file ABC123.DGN can appear once in Project_1, Project_2, Project_3, etc., all the way up to Project _100, if necessary.

Next, let's consider what a database with a filename uniqueness level of 3, which is a Primary Index Key of 'DirLevel1;DirLevel2;Filename', permits in terms of duplicate filenames.

With a filename uniqueness level of 3, ABC123.DGN can appear once anywhere at or below each branch of the 2nd level subdirectories in each project. For example:

❏ 15207 is the project, or DirLevel1, directory.
❏ d15207, b15207, and s15207 are 2nd level, or DirLevel2, subdirectories.
❏ Each 2nd level subdirectory continues to branch off to different levels.
❏ ABC123.DGN can appear once in each 2nd level subdirectory or anywhere within a branch of a 2nd level surdirectory.

With a filename uniqueness level of 4, ABC123.DGN can appear once anywhere at or below each branch of the 3rd level subdirectories in each project, as well as in any directory at the 1st level and any subdirectory at the 2nd level, and so on. In effect, as you increase the level of filename uniqueness, you also broaden the range of directories in which duplicate filenames can occur, until, at filename uniqueness level 7, the same filename can appear in any directory throughout the data structure.

The Data Repository / File Server

It is likely that you already have at least one server acting as a data repository. If you have one or more data repositories, it is probable that you already maintain electronic documents in these project directories. We refer to these existing documents as 'legacy data'.

It is important to note that Falcon/DMS does not limit you in the number of data repositories, nor are you limited to a single operating system—recall that Falcon/DMS supports Novell and UNIX, in addition to the Microsoft Windows Network OS. (However, as indicated earlier, the demonstration installation workflow concentrates on the Microsoft Windows Network OS.

There are several guidelines that govern how your data must be organized on a data repository:

❏ A single project cannot 'span' servers. In other words, all the files for a project must be maintained on a single data repository.
❏ Your directory structure cannot exceed a depth of six (6) layers.
❏ Maximum filename length is 80 characters, including spaces.
❏ Maximum directory name length is 40 characters, including spaces. (Directory name length refers to the length of each component of a project path.)

[Perform Step 1 from either the Control Panel's ODBC Data Source Administrator, or from any ODBC tools installed with the your database's client components.]

Step 1

Define the ODBC Data Source Name. (Refer to Appendix A for specific information on configuring a data source):

To define a data source you will require the database server's name or IP address, and the login ID and password for the portion of the database reserved for Falcon/DMS.

The data source you define in this step will be also be used in the Administrator's Database Account form in Step 2. In fact, the data source defined for all workstations must match the data source indicated in the Administrator.

Typically, the data source used for Falcon/DMS is named **FALCON**. This also happens to be the default data source assigned in the Database Account form. If you choose to use a different name, you will have to modify the data source entry in the Database Account form (see Step 2).

[Perform Steps 2-5 from the Falcon/DMS Administrator.]

Step 2

Set the Data Source and the database login account and password:

When Falcon/DMS in installed, a default data source, login account, and password are defined in the Administrator's Database Account form. These values are:
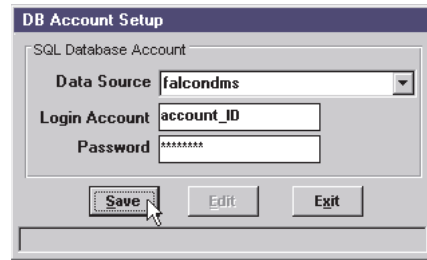
❏ Data Source: falcon
❏ Login Account: falcon
❏ Password: tsa

It is likely that these default values, certainly for the login ID and password, will not match the values assigned to the ODBC data source and to the actual database login account and password. The procedure outlined below assumes that the ODBC data source is named falconDMS rather than the default (falcon).

1. From the Start button, select Programs > Falcon_DMS > Administrator, or in the Falcon_DMS program folder, double-click the Administrator icon.

2. A message box opens, indicating that a data source name cannot be found. Press OK. The Falcon/DMS Administrator opens.

3. On the Falcon/DMS Administrator, select Database > Create Database. The Create Database dialog box opens.
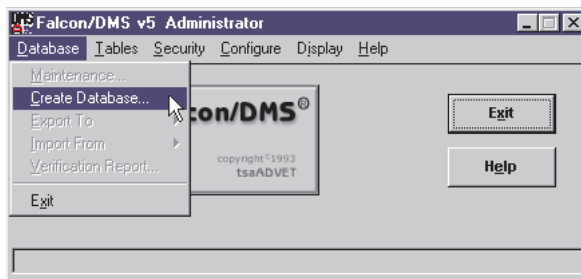
4. On the Create Database dialog box, press Set Account. The Set Account dialog box opens.

5. On the Set Account dialog box, press Edit and update the fields as needed. Press Save and then Exit.

DB Account Setup

SQL Database Account

Data Source falcondms

Login Account account_ID

Password ********

Save Edit Exit

Default Data Source: falcon—enter a valid Data Source: [e.g.] falcondms
Default Login Account: falcon—see database administrator

DB Account Setup

SQL Database Account

Data Source falcondms

Login Account account_ID

Password ********

Save Edit Exit
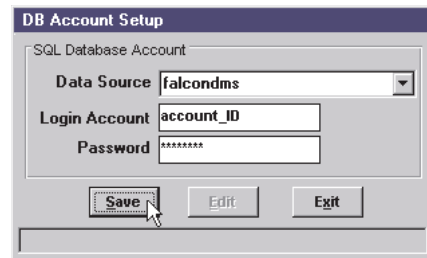
Default Password: tsa—see database administrator

6. A message box opens indicating that changing the account name may result in losing access to the database. Press OK.

If the ODBC data source is properly configured and a valid login account and password were entered in step 5, the Administrator will now be able to communicate with the portion of the database reserved for Falcon/DMS.

If a message box appears indicating that 'the login failed for [user]', have your database administrator verify the validity of the login account and/or password.

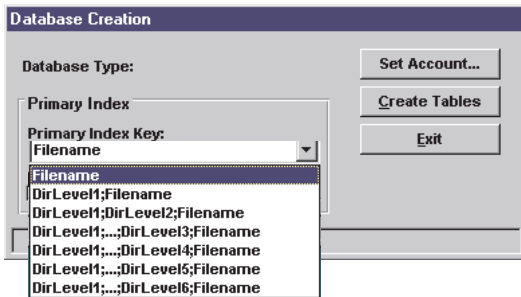The Database Creation dialog box becomes your active window.
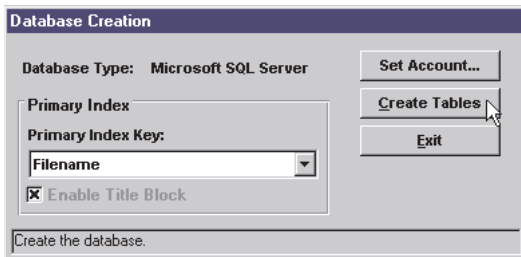
Proceed to Step 3.

Step 3

Define the database's Primary Index Key and create the Falcon/DMS tables:

"Do I want each filename to be used only once within my database, or, do I want to be able to use that filename more than once?" (See page 8—The Falcon/DMS Database Tables.)

1. On the Create Database dialog box, set the Primary Index Key.

```
Database Creation

Database Type:                        Set Account...

Primary Index                         Create Tables
Primary Index Key:
Filename                    ▼             Exit

Filename
DirLevel1;Filename
DirLevel1;DirLevel2;Filename
DirLevel1;...;DirLevel3;Filename
DirLevel1;...;DirLevel4;Filename
DirLevel1;...;DirLevel5;Filename
DirLevel1;...;DirLevel6;Filename
```

2. On the Create Database dialog box, press Create Tables.

```
Database Creation

Database Type:  Microsoft SQL Server     Set Account...

Primary Index                            Create Tables
Primary Index Key:
                                            Exit
Filename                        ▼
☒ Enable Title Block

Create the database.
```

Press Exit

3. The Administrator becomes your active window.

From the Administrator, select Database and note that the Create Database command has been replaced with Display Database.

Proceed to Step 4.

✔✔ HOT TIP

In the event you need to change the primary index after you create the tables, but before you create permanent records, request that the database administrator drop the table named FALCON50.

Than reset the Primary Index Key to the desired value and press Create Tables.

Please refer to Chapter 11, Database Maintenance, if you need to change the primary index after you create permanent records.

Step 4

Set the Falcon/DMS Network Configuration to UNC and define the server and resource tables:

The term 'seeding the database' is used to describe creating records for legacy data; this happens in Step 7 of Phase III. In order for Falcon/DMS to locate this legacy data, it has to be told, precisely, where on the network to look. As a preliminary to seeding the database, you will now provide the information needed to locate your legacy data and designate the method, or 'network configuration', by which Falcon/DMS uses this information. The network configuration used for this seeding process is UNC, or Universal Naming Convention.

When completed, this step enables Falcon/DMS to see shared folders on your data repositories—specifically, the shared folders under which you store your project directories.

To locate these project directories, and the files contained therein, Falcon/DMS uses 'server' and 'resource' tables that, respectively, list your servers' computer names (network node names) and the shared folders under which the project directories are maintained. This combination of server and resource precisely reflects the Universal Naming Convention of \\server\sharename.
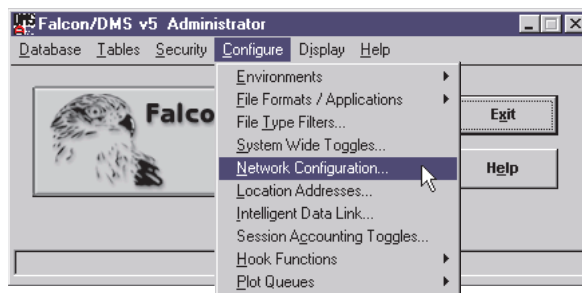
1. From the Falcon/DMS Administrator, select Configure > Network Configuration. The Network Configuration dialog box opens.
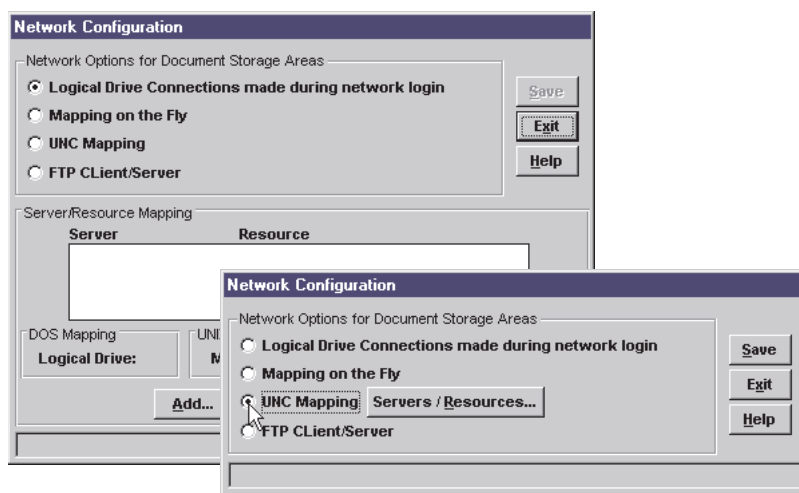
2. The default network configuration is Logical Drive Connections. Press the radio button for UNC Mapping.

   After the form re sizes, press Servers/Resources. The Network Tables dialog box opens.

3. In the Server input field, enter the data repository's computer/network node name.

After entering each server, press <Enter> or press Add. When the server list is complete, press Save.

Press the Resource List tab. In the Resource input field, enter the shared folders under which you maintain the project directories.

After entering each resources, press <Enter> or press Add. When the resource list is complete, press Save.

Press Exit. The Network Tables dialog box closes.
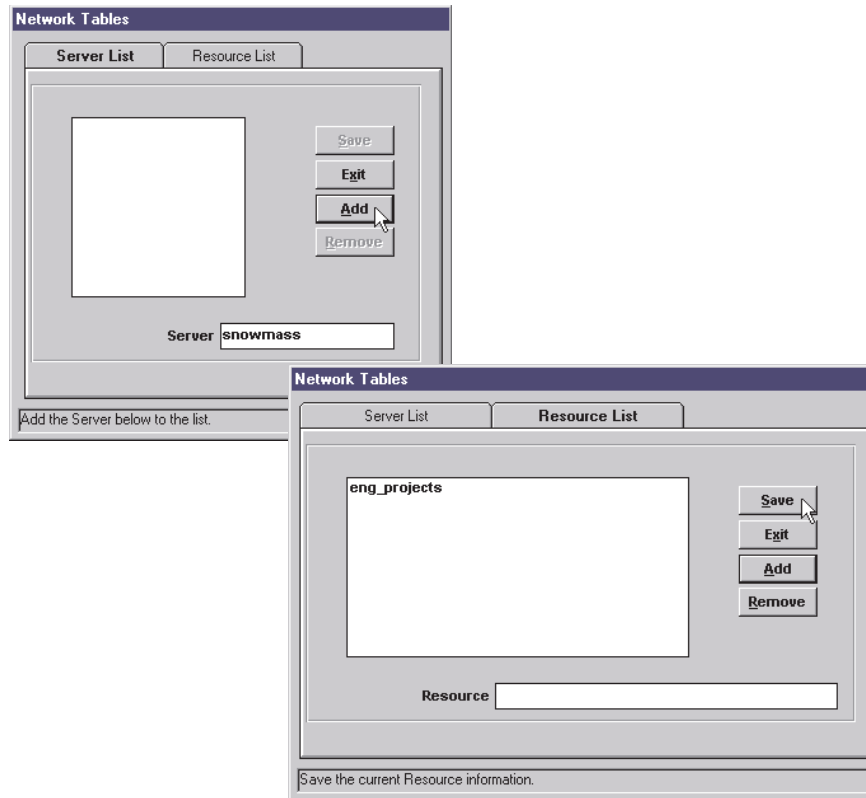
4. On the Network Configuration dialog box, press Save and Exit.

You are now ready to build the File Format Table. Proceed to Step 5.

Step 5

Define the File Format Table:

A File Format is an attribute assigned to a document's record as it is added to the database. Falcon/DMS uses this attribute to define parameters that control how certain operations behave and to identify which application opens the document, to name a just a few functions of the File Format. Falcon/DMS will not create a document's database record without a File Format.

When you open the File Formats/Applications dialog box you will see three tabs: File Formats, Application Properties, and File Format Mapping. For the initial implementation phase you will be concerned only with the File Formats tab. For the purpose of creating database records, File Formats do not have to be associated with applications, and File Format Mapping does not come into play at this point of an implementation.

What we need to accomplish in this step is to build the list of File Formats that will be associated with the documents we will 'bulk load' with MassAdd.
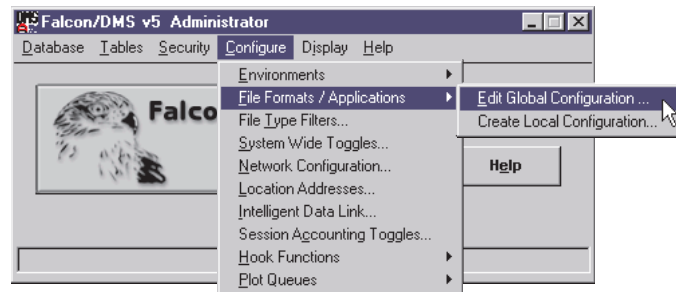
1. From the Falcon/DMS Administrator, select Configure > File Formats/Applications > Edit Global Configuration. The File Formats/Applications dialog box opens.
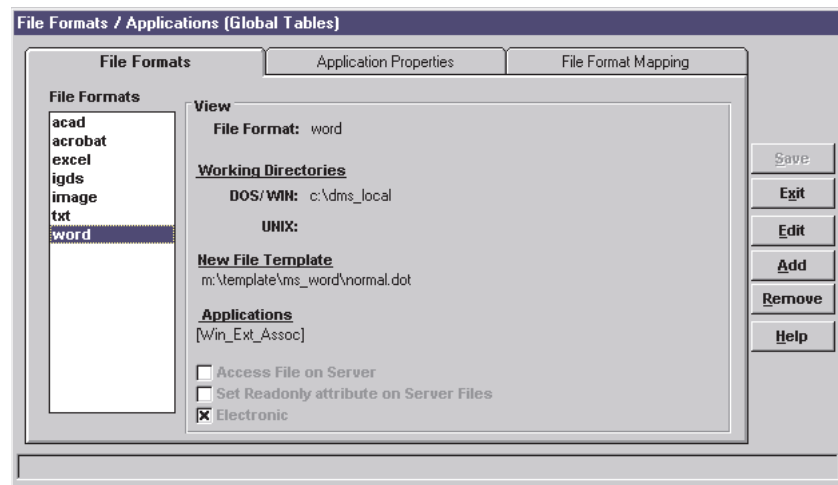

2. A set of pre-configured File Formats is delivered with Falcon/DMS. As you can see, the list includes both AutoCAD and MicroStation (acad and igds), several standard office automation formats (acrobat, excel, and word), and several non-standard formats (image and txt).

MassAdd uses this list to build its File Format drop-down list box. If you anticipate needing additional File Formats for the 'bulk load' phase of the database, proceed with the following steps.

3. Press Add.

    The View region of the File Formats tab changes to the Add region, input fields become available and several buttons appear.
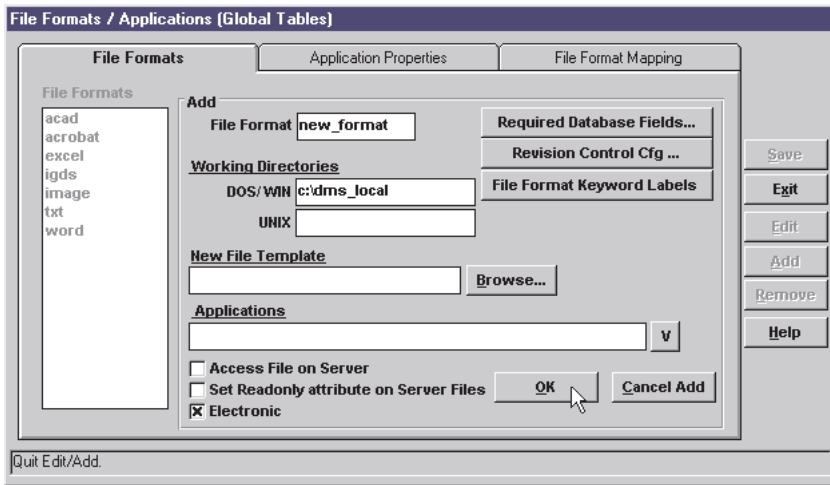
4. In the File Format field, enter a file format.

5. In the Working Directories > DOS/WIN field, define a local working directory. This is the directory to which Falcon/DMS copies files as they are opened or checked out. Because this field can be modified as you edit the File Format, a temporary working directory can be defined at this point.

6. Press OK, then Cancel Add.

7. Press Save, then Exit.

    You are now ready to build the Master 'Project' Table and begin the seeding of the database. Proceed to Step 6 on the following page.

Steps 6 and 7 are performed from the Falcon/DMS MassAdd Utility.

The MassAdd utility serves two purposes: In this step you will use MassAdd to define the Master 'Project' Table. This table lists the top-level directories on the data repository, the server containing the documents Falcon/DMS will control. The Master 'Project' Table is used by the Windows and MDL Document Managers to give users access to specific project directories on data repositories.

In Step 7 you will use MassAdd to bulk load the database. Bulk loading the database means that a single operation you create database records for files located in one, or several, or all 'project' directories, while associating each file's record with the appropriate File Format.
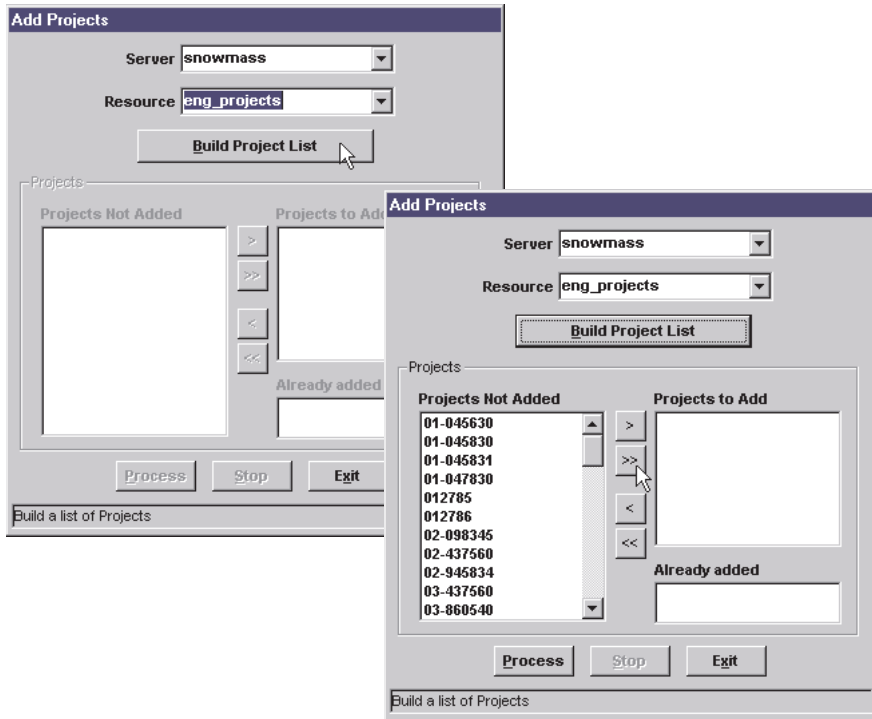
Step 6

Define the Master 'Project' Table:

1. From the Start button, select Programs > Falcon_DMS > MassAdd or, in the Falcon_DMS program folder, double-click the MassAdd icon.
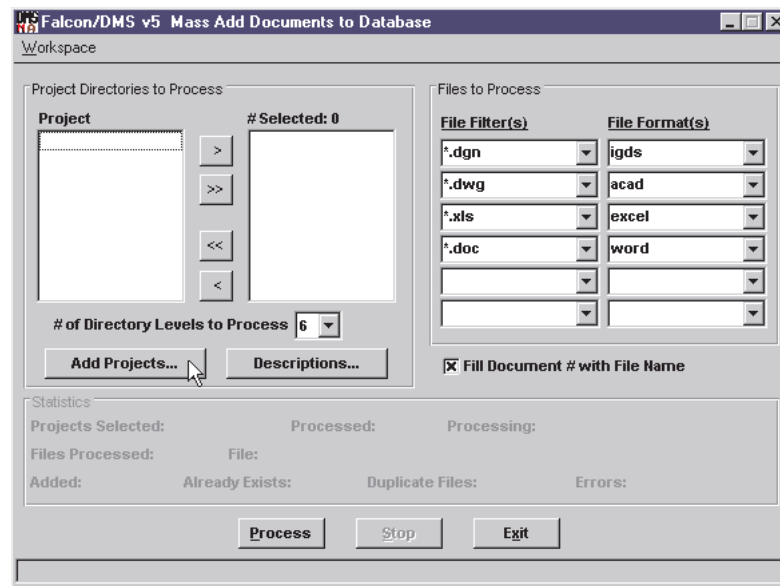
2. On the MassAdd form, press Add Projects... The Add Project dialog box opens.

3. From the Server drop-down list box, select the data repository from which Falcon/DMS will build a project list.

    From the Resource drop-down list box, select the shared folder from which Falcon/DMS will build a project list.

4. Press Build Project List.

5. From the Project Not Added list, select the projects you wish to add to the Master 'Project' List.

6. Press Process. When completed, press Exit.

    Repeat this process for different combinations of servers and resources, as needed.

    You are now ready to create database records for the documents found within these projects. Proceed to Step 7.

Step 7

From MassAdd, bulk load the database.

MassAdd assigns a File Format to each document as it is added to the database. In the File to Process region, you will define a set of up to six 'Extension (File Filter)—File Format' combinations. MassAdd scans each project directory for files that match these combinations. Ignoring documents with extensions not listed, MassAdd creates a database record for each document with an extension indicated in the Files to Process list.

When the list of specific file extensions exceeds six, run MassAdd again.

In the Files to Process scheme shown below, MassAdd will require only one pass of the selected projects. The specified extensions, or File Filters, cause files with those extensions to be added with the indicated File Formats assigned. Documents with any extension, other than the five specified extensions, are added to the database with the 'igds', or MicroStation, File Format.
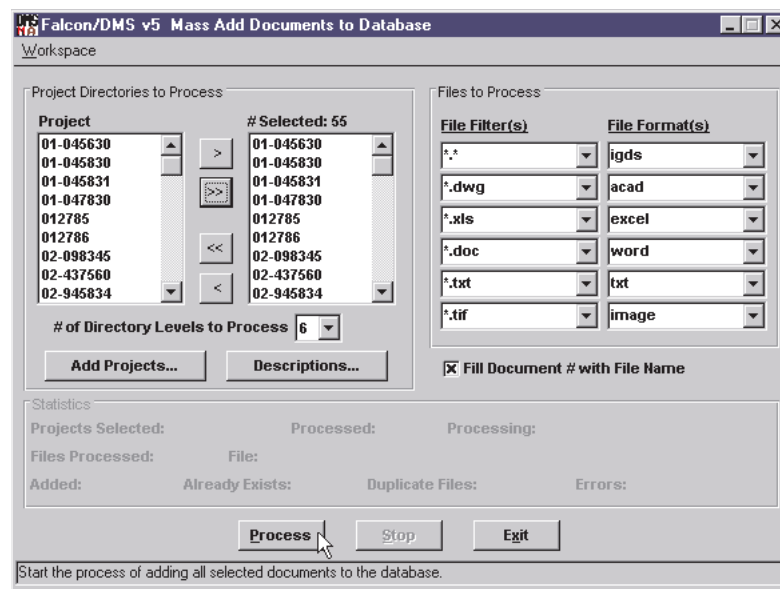
1. In the Project Directories to Process region, select the projects that MassAdd is to scan.

2. Also in the Files to Process region, use the Directory Levels to Process drop-down list box to indicated the subdirectory depth to which MassAdd scans for documents.

3. In the Files to Process region, define the necessary File Filter—File Format combinations.

   Enable the Fill Document Number with File Name check box to assign the filename, less the extension, to the Document Number database field for each new record.

4. Press Process.

   MassAdd will overwrite or append the latest results to a log file. MassAdd also provides running statistics—duplicate files are identified in MASSADD.LOG.

Phase IV

In Phase IV you will prepare the users' workstations and install the Falcon/DMS the Windows Document Manager and, as an option, the MicroStation (MDL) Document Manager.

Step 1

Define the ODBC Data Source Name. (Refer to Appendix A for specific information on configuring a data source):

To define a data source you will require the database server's name or IP address, and the login ID and password for the portion of the database reserved for Falcon/DMS.

The data source name is same as that defined for the administrator's workstation.

Step 2

On each user's workstation, define the required environment variables.

A minimum of two environment variables is required for Windows NT while a minimum of four is required for Windows 95 and 98 clients. The 'dcspcnod' and 'username' variables required for Windows 95 and 98 are the equivalent of the system-defined Windows NT variables, named 'workstation' and 'username'.

Modify the path variable only when the MicroStation component is installed locally.

Windows NT—a minimum of two variables is required:

1. From the Control Panel, select System > Environment

2. Define the following as *system* variables:

VARIABLE	EXAMPLE VALUE	DESCRIPTION
falcon_dir	[C]:\falcon5	This variable points to the local Falcon/DMS directory. The SETUPWIN.EXE program creates this local directory and several subdirectories, and delivers a small set of configuration files to this location.
falcon_sdir	\\server\dms_serv	This variable points to the server components on the application server. The example value shows a UNC path to the dms_serv directory on your application server. You may also use a mapped network drive as the falcon_sdir value
path	[C]:\falcon5\mdlapps	[Optional] The location of the local MicroStation components.

Windows 98/95—a minimum of four variables is required:

1. Open autoexec.bat.

2. Add the following variables [syntax: set variable = value]:

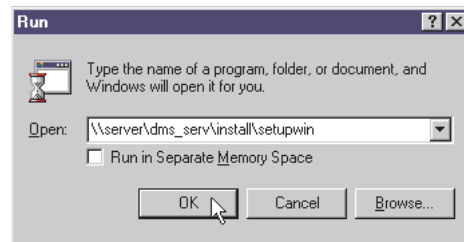| VARIABLE	EXAMPLE VALUE	DESCRIPTION
falcon_dir	[C]:\falcon5	This variable points to the local Falcon/DMS directory. The SETUPWIN.EXE program creates this local directory and several subdirectories, and delivers a small set of configuration files to this location.
falcon_sdir	\\server\dms_serv	This variable points to the server components on the application server. The example value shows a UNC path to the dms_serv directory on your application server. You may also use a mapped network drive as the falcon_sdir value.
username	your logon account	Required to audit transactions on the current workstation.
dcspcnod	computer name	Required to audit transactions on the current workstation.
path	[C]:\falcon5\mdlapps	[Optional] The location of local MDL components.

Step 3—optional:

Define a persistent mapped drive to \\server\dms_serv.

Step 4

1. If you chose not to map a drive to your application server, select Start > Run. In the Open field, key in the following and press OK:

   \\server\dms_serv\install\setupwin

   where [server] is your application server and [dms_serv] is the shared directory containing the Falcon/DMS server software.
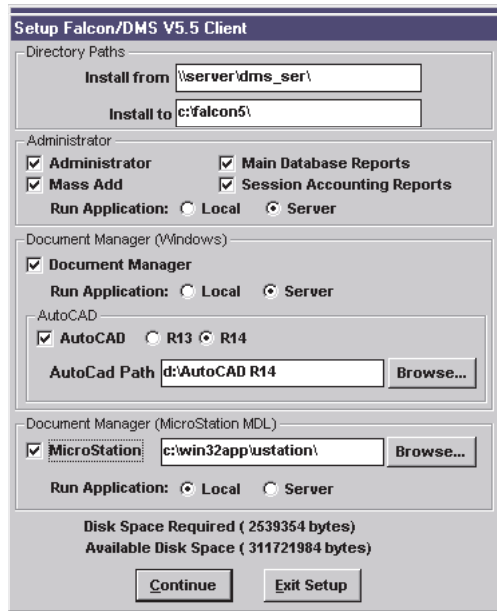

   If you have a mapped drive to dms_serv on your application server, open the Windows NT Explorer and navigate to the [M]:\install directory, where [M] is mapped to \\server\dms_serv.

   Double-click on SETUPWIN.EXE.

2. The Setup Options form opens; choose the components to deliver to the administrator's workstation.

   Setup options:

Setup Falcon/DMS V5.5 Client

Document Manager (Windows)—Select the Document Manager check box:

The Windows Document Manager is necessary to properly administer your Falcon/DMS installation. In addition to standard file management commands, the Document Manager provides the administrator with the ability to easily create document records, process files for Location Checkout and Check in, and Archive and Unarchive. Access to the Location and Archive commands is typically limited to administrators (see Chapter 13).

❏ Run Application:
   Local: Copies the Document Manager to your local hard disk.
   Server: Runs the Document Manager from the server.

AutoCAD (if required): when selected, this option modifies AutoCAD's main menu file (acad.mnu) and main lisp file (acad13/14.lsp), placing a pull–down menu on the AutoCAD menu bar that links to the Windows Document Manager. AutoCAD's File > Open and File > New commands are redirected to the Windows Document Manager when the main lisp file is modified.

❏ AutoCAD Path: press Browse to define the location of AutoCAD.
❏ Press the R13 or R14 radio button to identify the version of AutoCAD installed on the administrator's workstation.

Document Manager (MDL)[if required]—Select the Document Manager check box:

This option installs an MDL version of the Document Manager that is fully integrated with MicroStation's MDL interface.

❏ MicroStation Path: press Browse to define the location of MicroStation.
❏ Run Application:
   Local: Copies the Document Manager to your local hard disk.
   Server: Runs the Document Manager from the server.

Customizing the Administrator

As you work through the basic and advanced configuration issues of your Falcon/DMS implementation, you will find that you access certain dialog boxes more than others. To help you work more efficiently within the Administrator environment, you can customize the interface to display buttons that quickly open the most frequently used dialog boxes.
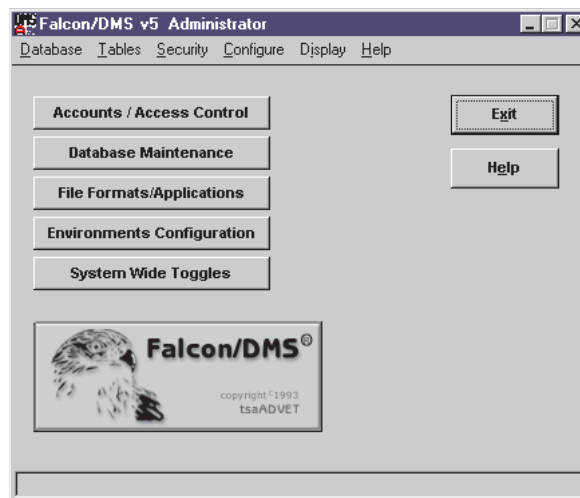
Defining Hot Buttons

1. On the Administrator, select Display > Define Menu Hot Buttons. The Menu Hot Buttons dialog box opens.
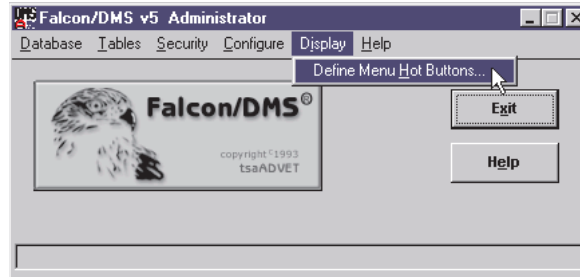
2. On the Menu Hot Button dialog box, double-click up to five items listed in the Available Options window.
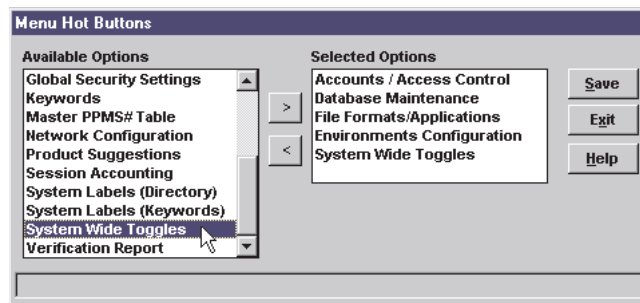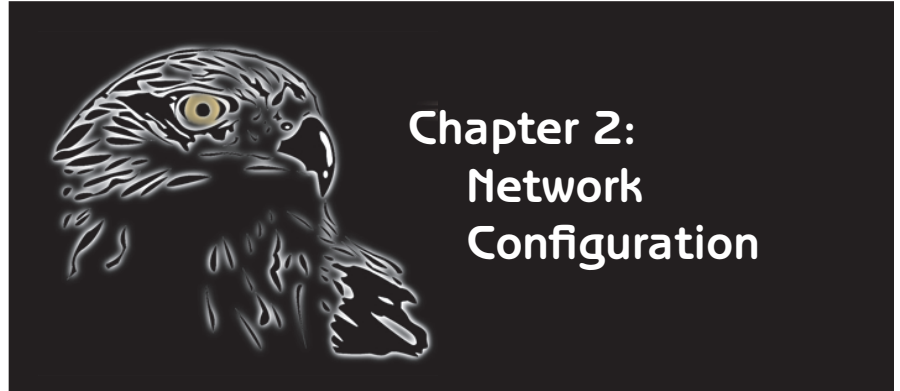
3. On the Menu Hot Button dialog box, press Save and then Exit.

Falcon/DMS

The Administrator

Chapter 2:
Network
Configuration

When Falcon/DMS is used to open or check out a document that resides on a data repository (an Intel-based or Unix server used to store files), it typically places a copy of that document (the 'working' document) in a predefined location on a client workstation, and modifies that document's database record to indicate where, when, and to whom the document is 'checked out'. Falcon/DMS maintains the repository version of this file (the 'record' document) as a 'read only' document until the working document is checked in. When the working document is checked in, it is removed from the client workstation and placed back on the data repository, overwriting the older record document. At this point, the new record document becomes available to authorized users for modification.

The focus of this chapter is on how Falcon/DMS uses the available network resources to accomplish the process described above.

As a basic guideline, Falcon/DMS is capable of managing documents on network operating systems that support mapped drives or FTP. Falcon/DMS can be configured to interface with a network OS with any of the following options: Mapped drives (logical drive connections); Mapping-on-the fly (a non-persistent mapped drive); UNC (Universal Naming Convention); FTP.

A logical drive connection, or mapped drive, allows a workstation to access documents stored on a server, as if those files were located on a physical drive on the workstation itself. However, because a logical or mapped drive permits users to view the connection to the data repository from the Windows Explorer or File Manager, the mapped drive option is not the preferred setting for the Falcon/DMS network configuration.

UNC and FTP, however, allow Falcon/DMS to perform operations on documents without a viewable connection to the repository server. Instead, UNC and FTP communicate with servers using procedures that are completely transparent to the user, and never establish a direct connection that can be viewed through the Windows NT Explorer or File Manager.

☞ **ATTENTION**
It is important to recognize that, as you define the Falcon/DMS network configuration, you are in no way changing your actual network conditions. You are building lists that Falcon/DMS uses to interpret the existing conditions.

➻ **TERMS**
Server Table: Lists the existing data repositories—the servers containing the files that are controlled by Falcon/DMS.

Resource Table: Lists the directories on the data repositories indicated in the Server Table. 'Resource' is a term used by Falcon/DMS that describes a Share Name on a Microsoft Windows Network, for example, or a volume and directory path on a Novell network.

✔ **TIP**
In addition to the Network Tables form, entries to the Server and Resource Tables can also be made directly from the Add /Modify Resource Mapping dialog boxes and from the Master 'Project' Table. Simply enter a server or resource in the appropriate field; if the entry does not exist in either table, Falcon/DMS will ask if you wish to add it. Press Yes to proceed.
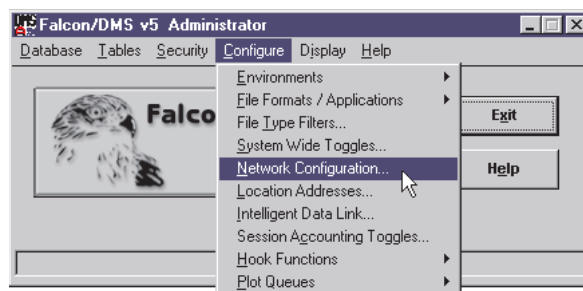
Network Resources

The four Falcon/DMS network configuration options use two tables to locate 'resources' on your repository servers. The first table, Servers, lists the actual data repositories, while the second table, Resources, lists the directories on the data repositories in which the data files controlled by Falcon/DMS are stored. Essentially, these two tables are used in combination to point Falcon/DMS to these specific storage areas. 'Resource' is a general term that Falcon/DMS uses to describe the specifications various operating systems use to list the server directories available to networked clients. For example, Windows and Unix use shared or exported directories, respectively, while Novell uses a specification referred to as 'volume & directory'.

These server and resource combinations are used by the MassAdd utility to build the original Master 'Project' Table and to create, as a bulk load process, database records for files found in the 'resource' directories, and for maintaining the Master 'Project' Table as the Falcon/DMS implementation expands to include new projects (see Chapter 4).

It is important to recognize that, as you define the Falcon/DMS network configuration, you are in no way changing your actual network conditions, rather, you are simply building resource tables that Falcon/DMS uses to interpret the those conditions.
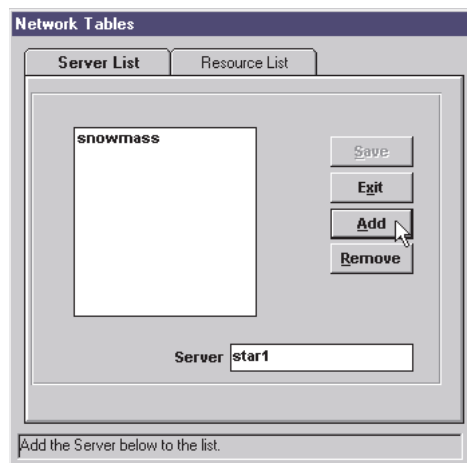
The Network Configuration dialog box

On the Falcon/DMS Administrator, select Configure > Network Configuration. The Network Configuration dialog box opens (opposite page).


### Network Tables: The Server & Resource Lists

All the Network Configuration options provide access to the Network Tables dialog box (pictured below). Use this form as one of two ways to add items to the Servers and Resource Lists as well as a way to remove any items no longer needed.

To open the Network Tables dialog box, press the Server/Resources button. This button displays on the Network Configuration dialog box when either Mapping on the Fly or UNC Mapping options is selected.
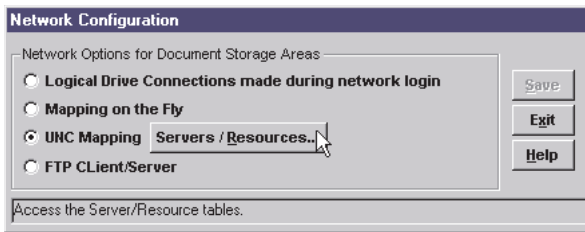
When either Logical Drive Connections or FTP Client/Server is selected, press Add to open the Add/Modify Resource Mapping dialog box—the Server/Resources button is now available is located on the Add/Modify Resource Mapping dialog box.
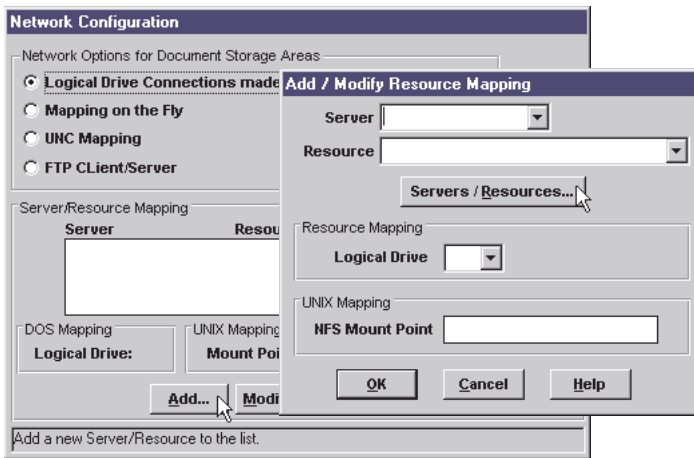
To Define a Server:

1. Select the Server tab, if required.

2. In the Server field, key in the server's Computer Name.

3. Press Add or press <Enter>. Repeat Steps 2 and 3 as needed.

4. Press Save and proceed to the next operation.

To Define a Resource:

1. Select the Resource tab.

2. In the Resource input field, enter a resource name. Recall that a resource name should correspond to a shared directory's actual Share Name, or the Novell and Unix equivalents. For example:

projects	Windows NT [Share Name] or
projects$	Windows NT [hidden Share Name]
vol1:projects	Novell Netware [(volume):(directory path)/(directory path)]
/usr2/projects	Unix NFS Server [/(Unix partition)/(directory path)]

3. Press Add or press <Enter>. Repeat Steps 2 and 3 as needed.

4. Press Save and Exit.

To Delete a Server or Resource:

1. Select the Server or Resource tab, as required.

2. In the Server/Resource list window, select a server/resource name.

3. Press Remove. Repeat Steps 2 and 3 as needed.

4. Press Save and Exit.

To Edit a Server or Resource:

1. Select the Server or Resource tab, as required.

2. In the Server/Resource list window, double-click a server/resource name.

3. In the key in field below the list window, edit the entry as required.

4. Press Save and Exit.

Mapped Drives—Logical Drive Connections made during network login

Logical Drive Connections, the first option on the Network Configuration dialog box, enables Falcon/DMS to use existing mapped drives. Because logical drives are usually mappings during a user's logon procedure, they are considered to be persistent and will be visible from the Windows Explorer or File Manager. If you wish to eliminate access to your documents from the Windows NT Explorer either FTP or hidden Share Names in combination with the UNC option.
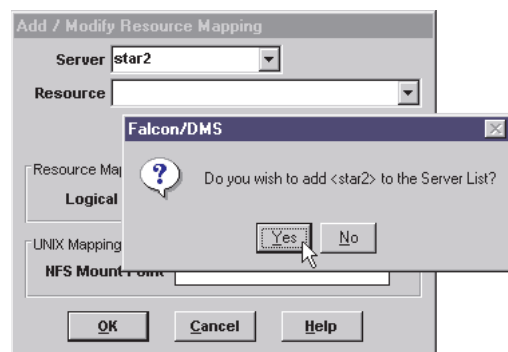
1. Press the Add button. The Add/Modify Resource Mapping dialog box opens.

2. Select a server from the Server drop-down list box.

   If the required Computer Name is not listed, press Servers/Resources and add the server's Computer Name to the Network Table.

   Note: Falcon/DMS also permits you to populate the Resource Mapping table directly from the Add/Modify Resource Mapping dialog box.

   In the Server field, key in the unlisted server's Computer Name—Falcon/DMS will ask if you wish to add the item. Press Yes.

3. Select a resource from the Resource drop-down list box.

   If the required Resource is not listed, press Servers/Resources and add the resource to the Network Table.

   Note: Falcon/DMS also permits you to populate the Resource Mapping table directly from the Add/Modify Resource Mapping dialog box.

   If the required Resource is not listed, key in the name of a shared directory using its Share Name—Falcon/DMS will ask if you wish to add the item. Press Yes.

If the required Resource is not listed, key in the name of a shared directory using its Share Name—Falcon/DMS will ask if you wish to add the item. Press Yes.

4. From the Resource Mapping region, select a drive letter. This letter must match the actual drive letter established for the clients that will use this resource. Press OK. Repeat as needed.

5. If the data repository is a Unix server, enter the full path specification of the exported directory in the NFS Mount Point input field.

6. On the Network Configuration dialog box, Press Save and Exit.

UNC Mapping

Universal naming convention (UNC) uses a name given to a device, computer, or resource to enable other users and applications to establish an explicit connection and access the resources over the network. The connection is not persistent and cannot be seen through the Windows NT Explorer or File Manager. The syntax of a UNC name is

\\<computer name>\<share name>\<directory> and/or <filename>

Of the two Falcon/DMS network configuration options recommended for local area networks, UNC Mapping is the easiest to implement. The only requirement is to list the computer names of your repository servers and their resources in the Network Tables.
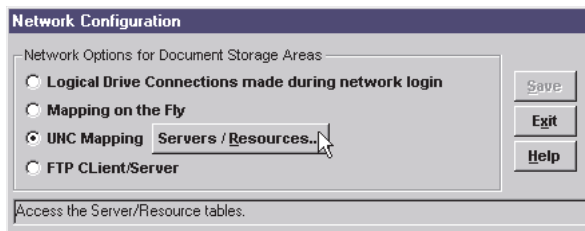
1. On the Network Configuration dialog box, select UNC Mapping.

2. Press Servers/Resources. The Network Tables dialog box opens.

3. In the Server input field, enter a server's Computer Name.

   Press Add or press <Enter>. Repeat as needed.

4. Select the Resource tab. In the Resource input field, enter a resource name.

   Press Add or press <Enter>. Repeat as needed. Press Save and Exit to close the Network Tables dialog box.

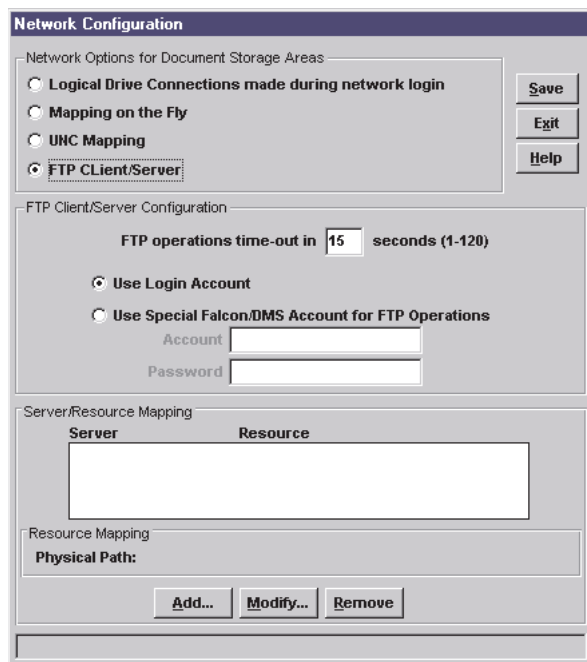5. On the Network Configuration dialog box, press Save and Exit.

FTP—Client/Server

File Transfer Protocol (FTP) is a language used to send or receive files over the Internet using a client/server architecture, and is the standard protocol for Unix networking operating systems. (Falcon/DMS uses standard FTP when communicating with a Unix data repository—no additional networking applications or services are required.) To use FTP with a PC-based network you must implement an FTP publishing service. (For Microsoft Windows Network, we recommend the FTP Server component available in Internet Information Server [IIS]. IIS is part of the Windows NT Server application. For Novell and other networking products, please consult your provider.)

Authenticating file & directory permissions on the data repository:

Special consideration must be given to choosing an Account option. To authenticate permissions on the data repository, the FTP Client/Server option can use each user's network Login Account or a special Falcon/DMS account.

❏ When the Login Account option is selected, Falcon/DMS relies on the explicit file permissions established through the operating system to authenticate permissions. This option also requires users to log in to the data repository each time they open Falcon/DMS.

❏ When the Special Falcon/DMS Account option is selected, the special account must own and have full rights on the directories and the files contained therein. The Login Account and Password you define from your network's account manager and the Account and Password settings defined in the Network Configuration's FTP form must be identical.

1. On the Network Configuration dialog box, select FTP Client/Server. The dialog box expands to show the FTP Client/Server Configuration options.

2. Specify an FTP operations time-out value. The higher the value (to a maximum of 120 seconds), the longer Falcon/DMS is given to establish the necessary FTP connections.

3. Choose an Account mode.

4. Press the Add button. The Add/Modify Resource Mapping dialog box opens.

5. Select a server from the Server drop-down list box.

   If the required Computer Name is not listed, press Servers/Resources and add the server's Computer Name to the Network Table.

   Note: Falcon/DMS also permits you to populate the Server and the Resource Mapping tables directly from the Add/Modify Resource Mapping dialog box.

   In the Server field, key in the unlisted server's Computer Name—Falcon/DMS will ask if you wish to add the item. Press Yes.

6. Select a resource from the Resource drop-down list box.

   If the required Resource is not listed, press Servers/Resources and add the resource to the Network Table.

   If the required Resource is not listed, key in the name of a shared directory using its Share Name—Falcon/DMS will ask if you wish to add the item. Press Yes.

7. In the Resource Mapping region, enter the actual path for the resource indicated in Step 6 in the Physical Path input field .

   Consider that shared or exported directories are often located several levels deep within a directory path. For example, the shared directory PROJECTS is located on a server hard drive designated as <F:>, under a directory named CADD. In this case the physical path is F:\CADD.

8. Repeat Steps 5, 6, and 7 as required. Press OK.

9. On the Network Configuration dialog box, press Save and Exit.

Running FTP and UNC together

Falcon/DMS's FTP network configuration does not allow you to create actual project directories and their subdirectories from the Master 'Project' Table dialog box. While you are able to add projects entries to the Master 'Project' Table when FTP is the network configuration, you cannot create directories on the server from the Falcon/DMS interface. In this version of Falcon/DMS a work–around is available to address this problem—however, this solution should be implemented only on a workstation used to modify the Master 'Project' Table. Typically, workstations used by the Falcon/DMS Administrator and any person responsible for creating projects will qualify.

By placing a copy of dmsneta.cfg, the file that identifies the network configuration, in the <falcon_dir>\cfg directory on a specific workstation, anyone logging on to that workstation will use this local configuration file, rather than the global configuration located on the application server. (A typical <falcon_dir\cfg directory is c:\program files\falcon5\cfg.) While the global version of dmsneta.cfg is set to FTP, this local version will be set to UNC, thereby allowing the Falcon/DMS Administrator interface to access the data repository.

We will explain, momentarily, two ways to place dmsneta.cfg locally. First, however, there is one very important issue that must be addressed before you change the network configuration for any workstation. The Document Manager, the user interface, relies on FTP to authenticate permissions on the data repository and perform any and all document transactions. As such, an environment variable must be defined on appropriate workstations so that the Document Manager continues to use FTP for document transactions. The environment variable that forces the Document Manager to use FTP is

   dms_network_mode = ftp. (Set from the Control Panel for NT and autoexec.bat for Windows 95 and 98.)

☞ ATTENTION

To use the UNC Mapping option on a network in which Windows clients are communicating with a Unix server, you will need to implement a Network File System (NFS) client-server application.

When this environment variable is defined and the network configuration defined in the local dmsneta.cfg is set to UNC, the Document Manager will continue to use FTP and the Administrator will now use UNC.

There are two ways to define a local version of dmsneta.cfg:

❏ Copy dmsneta.cfg from the application server to the local <falcon_dir>\cfg directory on the appropriate workstations (e.g. c:\program files\falcon5\cfg). The server version of dmsneta.cfg is located in the <falcon_sdir>/cfg directory (e.g. \\sw5009\dms_serv\cfg). Once dmsneta.cfg has been copied, the Administrator's Network Configuration environment reads the local version rather than the server, or global, version.

Then, from the Administrator, select Configure > Network Configuration. On the Network Configuration dialog box, change the current configuration from FTP to UNC. Press Save and Exit. The workstation now uses UNC for any operations using the Master 'Project' Table.

❏ Launch MassAdd—MassAdd automatically writes a temporary dmsneta.cfg, set to UNC, to the local <falcon_dir>/cfg directory. (The file is removed when MassAdd closes.) From the Windows Explorer, set the current directory to <falcon_dir>/cfg, then copy and paste dmsneta.cfg—the directory now contains dmsneta.cfg and copy of dmsneta.cfg. Close MassAdd; dmsneta.cfg is removed. Rename copy of dmsneta.cfg to dmsneta.cfg.

You are now able to use the Master 'Project' Table to create project directories and their subdirectories on the data repository.

PLEASE NOTE—Before creating any project directories through the Master 'Project' Table: If FTP uses the 'special account', you must be logged onto the local workstation under the same account name as that used for the FTP special account, otherwise, directories will not be created with the proper ownership. (See ✔ TIP)

Mapping On The Fly

This option is a holdover from the Windows 3.1 and Windows for Workgroups era and is considered to be outmoded.

Mapping on the Fly, the second Network Configuration option, provides a mapped drive that is active only when the Falcon/DMS Document Manager is open; when the Document Manager is closed the mapped drive disconnects.

Be careful not to assign a Primary or Secondary Drive letter that is already in use.

1. On the Network Configuration dialog box, select Mapping on the Fly.

2. Define your Servers and Resources as needed.

3. From the Network Software drop down menu, make the appropriate selection.

4. From the Primary Drive and Secondary Drive drop-down list box, select drive letters that are not in use by your network clients. Two drive letters are required for transactions such as Move and Copy documents.

5. Press Save and Exit.
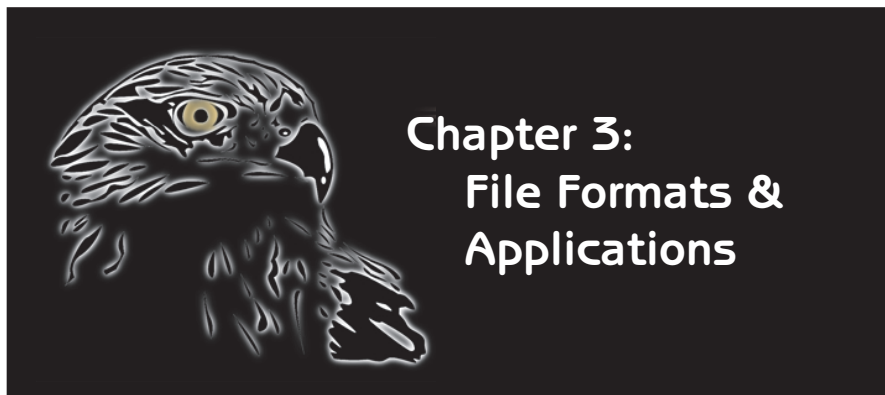
Falcon/DMS

The Administrator

Chapter 3:
File Formats &
Applications

The File Format attribute assigned to each document record as it is added to the database indicates which application, or in certain cases, which applications are used to open that document. However, the application association is only the most obvious function of the File Format attribute.

Because there are so many critical parameters associated with this attribute every document's database record must include a File Format. It is for this reason that defining the File Filter to File Format mappings is central to using MassAdd when bulk loading a database.

Even a non–electronic document has a File Format. For example, paper documents can be represented in Falcon/DMS by database records that might indicate the drawer in a plan file in which a set of paper documents is stored. It is a File Format attribute that tells Falcon/DMS what documents are not electronic, and therefor not under its direct control, and which documents are electronic.

Before we consider the many other parameters defined in this environment, lets examine the application function of the File Format.

Falcon/DMS supports virtually all Windows applications and is DDE compliant. This means that your organization can use the Falcon/DMS Windows Document Manager to manage the majority of Windows document, and automatically launch the application necessary to open that document. In practice, the File Format's application function is similar to that of the 'Open With' feature or the 'File Type > Actions' functions found in Windows.

Falcon/DMS actually takes this idea a step further by allowing multiple applications to be associated with a File Format. When this occurs, the user chooses which application shall open a document. This is particularly useful in the case of the MicroStation and AutoCAD programs. A feature of both is the ability to support different configurations of the same application. For example, an architectural configuration may differ significantly from a civil engineering configuration, which in turn differs from a mechanical configuration. However, Falcon/DMS can associate all three configurations with the MicroStation File Format. This means that, when you select a document,

☞ **ATTENTION**

MassAdd—For the purpose of bulk loading legacy data, a File Format does not have to be linked to an application; it is only when users are ready to open files with the Falcon/DMS Document Manager that the File Format needs this association.

✉ **NOTE**

A file format's application and the application's command line may be modified at anytime without affecting the integrity of the document's database record.

☞ **ATTENTION**

MicroStation's and AutoCAD's basic application properties and those of a selection of popular Windows-based applications are predefined; it is possible that these properties will have to be modified to conform to your environment.

✔ **TIP**

You can double-click on items in the list windows to automatically invoke the 'Edit' operation.

the Document Manager allows you to choose which of the three 'flavors' of MicroStation is appropriate for that design session. (Refer to Chapter 11 for a detailed explanation of the File Formats & Applications controls for MicroStation and AutoCAD.)

Falcon/DMS takes this level of flexibility even higher by making the File Format functions available to the Multiple Environments control, giving you the ability to provide different File Format schemes for Falcon/DMS configurations tailored to different groups, departments, or individuals. (see Chapter 6).

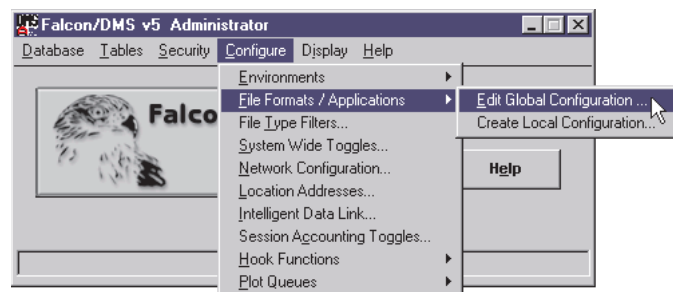In summary, the File Formats/Applications environment is used to:

❏ Define and edit a file format.
❏ Define or edit an application and its properties.
❏ Associate an application or applications with a file format.
❏ Define the local directory into which documents are placed when Falcon/DMS opens or checks out a file. (An application's local directory is called its 'working' directory.)
❏ Define the database fields that must be filled in when certain operations are performed—for example, you can require that users provide a description when a CADD file is created, or a word processing file is added to the database.
❏ Configure Revision Control.
❏ Define the File Format Keyword Categories and Keyword Templates.
❏ Establish additional parameters specifically for MicroStation and AutoCAD.
❏ Define File Association parameters.
❏ Define File Format Mapping parameters.

Defining File Formats and Application Properties

The File Formats/Applications dialog box is used to establish the parameters that determine how Falcon/DMS will handle different document types, and to define the properties for the applications used to create and modify those documents.

To Open the File Formats/Applications dialog box

From the Administrator, choose Configure > File Formats/Applications > Edit Global Configuration. (The Create Local Configuration option is discussed later in this chapter.)


The File Formats/Applications dialog box is comprised of three tabs:

❏ File Formats
❏ Application Properties
❏ File Format Mapping

Let's start with the File Formats tab.

The File Formats tab

The File Formats tab is used to add, edit, and remove File Formats. The tab lists the available File Formats and provides access to various input fields, check boxes, and dialog boxes used to assign their parameters.
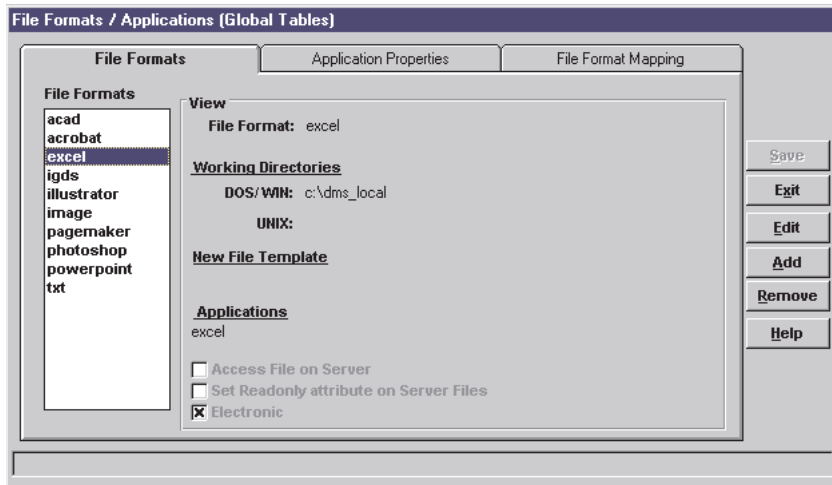
As mentioned earlier, because there are so many critical parameters associated with this attribute every document's database record must include a File Format.

The most obvious is the association of a File Format with an one or more applications. Beyond application associations are the parameters that determine how users access documents and interface with their database record. For example, the 'Working Directory' parameter determines whether Falcon/DMS delivers a document to the requesting workstation or to a 'username' directory on a server; the 'Access File on Server' check box indicates if the document remains in its home directory on the data repository and is 'Reserved by user' instead of checked out to a workstation.

There are three File Format controls discussed in detail in other chapters.

Several predefined File Formats are delivered with Falcon/DMS. Two of these, the CAD File Formats for MicroStation (igds) and AutoCAD (acad), provide an additional set of options unique to these applications and are discussed in detail in Chapter 11. As shown below, when 'igds' (the MicroStation File Format) is selected, a button labeled 'Additional igds Parameters' appears in the Edit region. Similarly, when 'acad' is selected, an 'Additional acad Parameters' button appears.
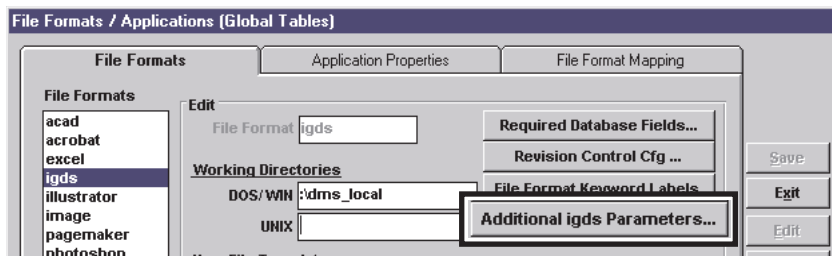
The two remaining File Format controls discussed in other chapters are 'File Format Keyword Labels' (Chapter 9) and 'Revision Control Cfg' (Chapter 12).

To add a File Format:

1. On the File Formats tab, press Add. The File Formats tab reformats itself such that the 'View' region becomes the 'Add' region.

2. In the File Format input field, key in a File Format name. While this name can be anything your wish, this explanation will use a standard Windows application. For example:

   word

The 'Working Directory' input fields are used to specify a base directory on the client workstations to which Falcon/DMS will copy requested files, while recreating their server directory paths. If you utilize Windows and UNIX operating systems in your environment, specify a working directory for each. In order to consolidate all files checked out through Falcon/DMS, no matter what their File Formats, consider specifying the same local working directory for all formats.

3. [Required] Press <Tab>. In the appropriate 'Working Directory' input field, enter a local working directory—for example:

   c:\dmslocal

4. [Optional] Press Browse or enter a directory location and filename in the 'New File Template' field. When a template file is designated for a File Format, the Document Manager's File > New command will permit you to create a file and its database record from the Windows Document Manager for that File Format.

5. Press the Application [V] button to open the Select Applications list box. Because this is a standard Windows application you may use the association defined in the Windows registry.
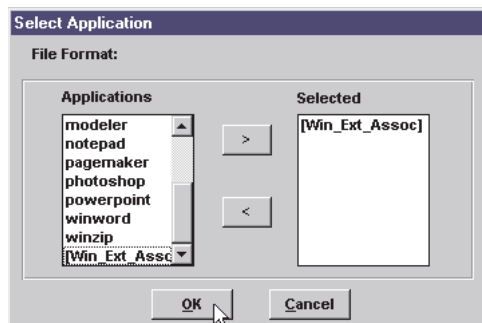
In the Applications list, double-click on [Win_Ext_Assoc]. Press OK.

To associate multiple applications with a File For)mat, double-click the appropriate applications to build a 'Selected' list containing two or more entries. The [Win_Ext_Assoc] option can be included in this list.

6. Define other options as required. When finished, press OK, Cancel Add, and Save.

To edit a File Format:

1. Double-click on an item in the File Format list window, or select an entry and press Edit.

2. Edit as needed.

3. When finished, press OK, Cancel Edit, and Save.

To remove a File Format:

1. Select an item in the File Format list window.

2. Press Remove. On the message box, press OK.

3. Press Save.

File Format options

CONTROL	DESCRIPTION
File Format	[input field] Enter a File Format name - 12 characters
Working Directory	[input field] The local directory to which Falcon/DMS checks out documents and, typically, replicates their server directory paths. The replication of the server directory paths is automatic when the database's Primary Index Key is greater than <Filename>. If you wish to replicate the server directory paths when the Primary Index Key is set to <Filename>, enable the 'Create Local Master/Subdirectories on checkout' control in System Wide Toggles (see Chapter 5).
	Working Directory Options -- You have three options for defining a working directory: 1) Check out documents and replicate theirserver directory paths on each workstation's hard disk. For this approach you can specify a different target path for each File Format or assign the same target path to all (for example, C:\DMS_LOCAL). 2) Check out documents and replicate their server directory paths to an area on a server defined by %username% variable. For example, P:\%USERNAME%, where P: is a mapped drive established for each user's login procedure. 3) Use Multiple Environment to assign File Format working directories to different groups, departments, or users (see Chapter 6).
New File Template	[input field or Browse button entry] Directory location and filename of template for documents created from the File > New command on the Windows Document Manager.
Applications	[list box] Used to select the application that is automatically launched when a document is opened from the Windows Document Manager. The applications and their properties are defined on the Applications tab. The list box contains [Win_Ext_Assoc], the application option that uses the application association defined in the Windows registry. For additional information on the Application options for MicroStation and AutoCAD, refer to Chapter 11.
Access File on Server	When enabled, files are not copied to a workstation's local hard drive, rather the file is accessed across the network. When a file is accessed on the server the database availability field reads 'Reserver By User'.
Set Read Only Attribute on Server Files	When enabled, files checked out through Falcon/DMS can only be accessed from outside Falcon/DMS as read_only. When Access File on Server is selected, this option is disabled.
Electronic	When disabled, the File Format describes a non-electronic document. This feature permits you to maintain records for non-electronic documents in the Falcon/DMS database. (See Non-Electronic Document Control, this chapter.)
Required Database Fields...	From the Required Database Fields form the Falcon/DMS administrator identifies the database fields that must be filled in when certain operations are performed (See Required Database Fields, this chapter.)
Revision Control Cfg...	Revision Control schemes can vary between file formats and can be enabled for some and not for others. Revision Control can be configured to create a new revision at file checkout or file checkin, either automatically or at a prompt. Revision Control can also be configured to allow users to create a revision at their own discretion. (See Chapter 12.)
File Format Keyword Labels	A keyword is a manager or user-defined attribute that facilitates database queries. Keywords can be assigned to six manager-defined 'system' keyword categories or 20 manager-defined 'file format' keyword categories. (See Chapter 9.)
Additional 'CAD' parameters	Provide an additional set of options unique to MicroStation and AutoCAD. (See Chapter 11.)

Required Database Fields

From the Required Database Fields dialog box you are able to designate the database fields that must be filled in when certain operations are performed.

For example, when the Create operation and the Document Number and Description attribute fields are enabled for a File Format, a user creating a document for that File Format must also fill in the description and document number fields before Falcon/DMS creates the file and its database record.
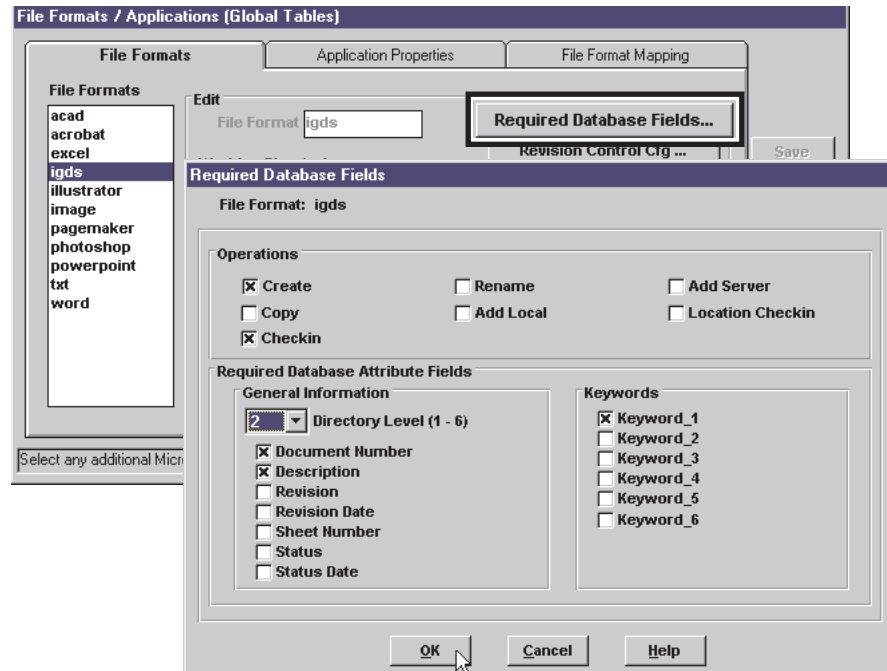
To define required database fields:

1. Double–click a File Format.

2. In the Edit region, press Required Database Fields. The Required Database Fields dialog box opens.

3. In the Operations region, select the operations that when performed, will require the user to enter information in the database fields checked below.

   In the example, users creating and checking in MicroStation (igds) documents will be required to fill in the Document Number and Description fields, and assign a keyword to the first of the six system-level keywords.

   The Directory Level (1-6) control indicates the highest directory level in which users can create or add documents, or to which users can move or copy documents. In the example, users will be restricted to creating or otherwise placing MicroStation design files in directory levels 2 through 6.

   When you create a new File Format, this control is automatically set to 1.

4. Press OK.

The Application Properties tab

The Application Properties tab is used to add, edit, and remove Applications and their properties. File Association, a feature available to each application, is also defined from this environment.
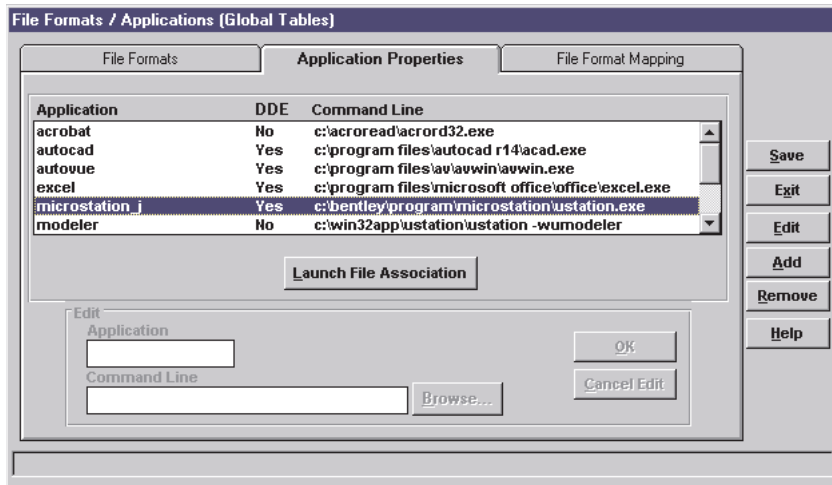
When you choose a file in the Windows Document Manager, Falcon/DMS reads the file's File Format attribute, which in turn identifies the application the Document Manager must launch in order to open the selected file. As you have seen, the File Format can be associated with either the [Win_Ext_Assoc] option, which reads the registry to identify the appropriate application, or one or more applications defined in the Application Properties environment.

As an example of the latter, this section demonstrates how to define the properties for two standard applications for which Falcon/DMS does not use a [Win_Ext_Assoc] option: Microsoft Paint, for editing raster images, and the Wang Image Viewer, for viewing raster images (both applications are delivered with Windows). The demonstration then defines a File Format called 'image', to which we will associate both applications. Finally, we will show how the Document Manager allows you to choose which application to launch when an 'image' document is opened.

To add an Application:

1. Select the Application Properties tab.

2. Press Add.

3. In the Application field, key in:

   ms_paint

4. In the Command Line input field, key in

   C:\WINNT\system32\mspaint.exe

   Or press Browse. From the Select Application Command Line dialog box, locate and double-click on MSPAINT.EXE. Press OK.

4. In the Application field, key in:

   wang_viewer

5. In the Command Line input field, key in

   C:\Program Files\Windows NT\Accessories\ImageVue\wangimg.exe

   Or press Browse. The Select Application Command Line dialog box opens. Locate and double-click on WANGIMG.EXE. Press OK.

5. Press Cancel Add and Save.

To associate MS_PAINT & WANG_VIEWER with a File Format:

1. Select the File Formats tab. Press Add.

2. In the File Format input field, key in:

   image

3. In the DOS/WIN Working Directory input field, key in:

   c:\dms_local

4. Press the Application [V] button to open the Select Applications list box. Because image is not a standard File Format you will not use the [Win_Ext_Assoc] option.
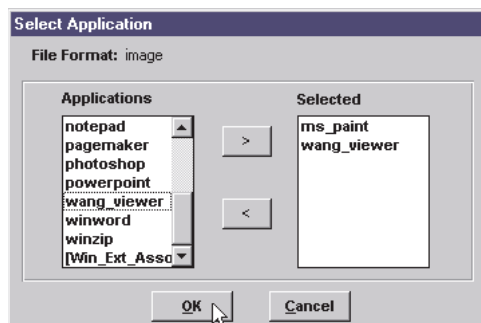
In the Applications list, doube-click on ms_paint and wang_viewer. Press OK.

6. Press OK, Cancel Add, and Save.

What the user sees
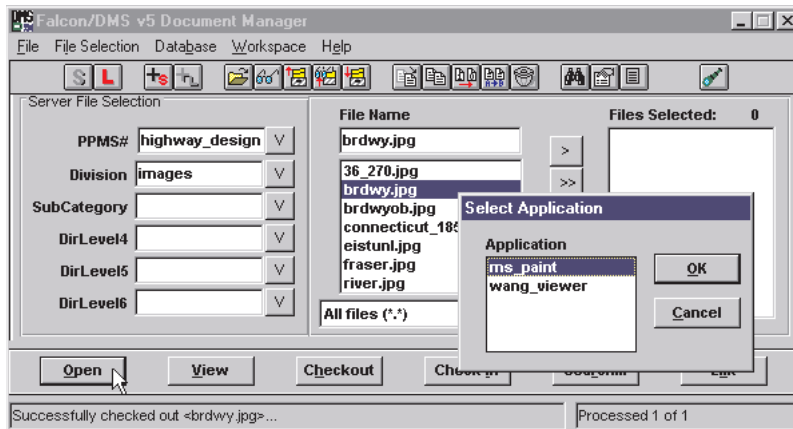
Under this configuration the Windows Document Manager now enables the user to choose which application to launch when an 'image' file is opened or copied out for viewing. Shown below is the Select Application dialog box that displays when a user selects either Open or View after selecting the 'image' file.

The procedure outlined above is similar to that used for CAD applications where different configurations of MicroStation or AutoCAD are available to designers. The benefits to assigning multiple applications to the 'igds' and 'acad' File Formats, and the specific procedures to follow for configuring this feature are detailed in Chapter 11.

To edit an Application:

1. In the Application list, double-click an application, or select an application and press Edit.

2. Edit the Application and Command Line input fields as required.

3. Press OK. Repeat as needed.

4. Press Cancel Edit and Save.

To remove an Application:

1. In the Application list, select an application.

2. Press Remove. Repeat as needed.

3. Press Save.

File Association

Within the engineering and architectural disciplines there is broad acceptance, indeed reliance, on application extensions for design software such as MicroStation and AutoCAD. This increase in the use of application extensions means that managing documents associated with a project, or a specific document within a project, has become proportionally more complicated. The reason—a typical engineering or architectural application extension constantly generates new files and updates existing files as a design advances from one phase to the next. When the challenge of managing this continuously changing set of documents falls mainly on the designer, the risk of losing track of which auxiliary files belong with what design document increases as the size of the dataset grows. To relieve the designer of this burden, Falcon/DMS now supports a feature that tracks new files created by application extensions and, with minimal user intervention, adds them to the Falcon/DMS database; this feature is called File Association. Through File Association, users are provided with a list of pertinent documents from which they can select as few or as many as required, eliminating the guesswork that is inevitable in any manual effort to manage large numbers of files. File Association places no limit on the number and type of 'child' files for which you can establish and manage associations, nor are you restricted in the type of file that can be designated as a 'parent' document.

When Falcon/DMS opens a document for which associations exist, the user may choose which auxiliary files, if any, are to be checked out with this 'parent' document. (File Association also permits the designer to choose which auxiliary files are checked out for modification, and which are copied out for reference.) When the original document is checked in at the end of a session, Falcon/DMS provides a list of any new files generated by the application extension and allows the user to choose which new files are to be added to the database and moved to the data repository, and which are to be discarded.

File Association is triggered by the Open and Check In commands, and is controlled by parameters defined for the application used to open the 'parent' document. Recall that each document in the database will always be assigned a File Format, selected from a list defined in the Administrator's File Formats & Applications environment; from this environment you also define applications and their properties and assign one or more applications to each File Format. The Application Properties tab on the File Formats & Applications dialog box is also where you enable File Association and define the rules that control the 'automatic association' of new files. It is important to understand that this is done on a per application basis. A detailed explanation of this is provided later.

Presently, File Association is used almost exclusively for associating files to CADD documents; for this reason, the examples presented in this section will rely on Bentley System's MicroStation and InRoads, Intergraph's civil engineering application extension, to first, illustrate the benefits of File Association, and then, to explain how File Association is configured. (However, recall that you are not restricted in any way in the type of files that you can define as 'parent' or 'child' documents.)

What the user sees

Before explaining how to set up File Association, it will be helpful to understand what the user sees when opening a document for which associations are defined, how newly generated files can be associated as part of the check in process, and how newly generated files can be associated as part of the check in process. (Please refer to the Document Manager guides for information on viewing associations and removing associations between documents.)

To open a document with existing associations:

When associations for a selected document have already been established and that document is opened by the application for which File Association rules have been defined, the user is presented with a form from which he can choose which auxiliary files are to be checked out as the selected document is opened. When associated file are checked out, the Availability for these files is lists as 'Associated'. As with any Open operation, the user first select a file; in this case the file is a 'parent' document:

1. From the File list window, the user selects a MicroStation design file and presses Open.

2. Before advancing to the next step, Falcon/DMS processes any reference files. When multiple applications are assigned to a File Format—'igds' in this case—the Select Application dialog box opens. In the example, the user will choose the InRoads option.

   Typically, when an application extension is installed for MicroStation or AutoCAD, Falcon/DMS will be configured to allow the user to launch either a 'vanilla' version of the CADD application, or a version configured to run with the extension. For this explanation, the associations seen in Step 3 were established for the InRoads flavor of MicroStation.
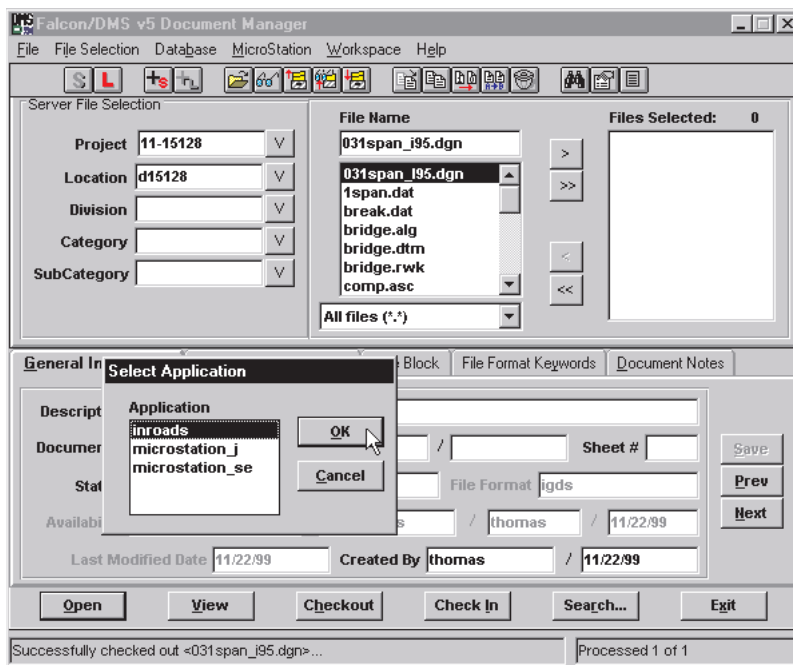
3. Because associations exist for the selected design file and the user has chosen the application for which the associations were established, Falcon/DMS next opens the File Association dialog box. This form presents a list of the 'child' files associated to the selected design file. Here the user chooses the files he wishes to check out for modification or to copy out for reference.
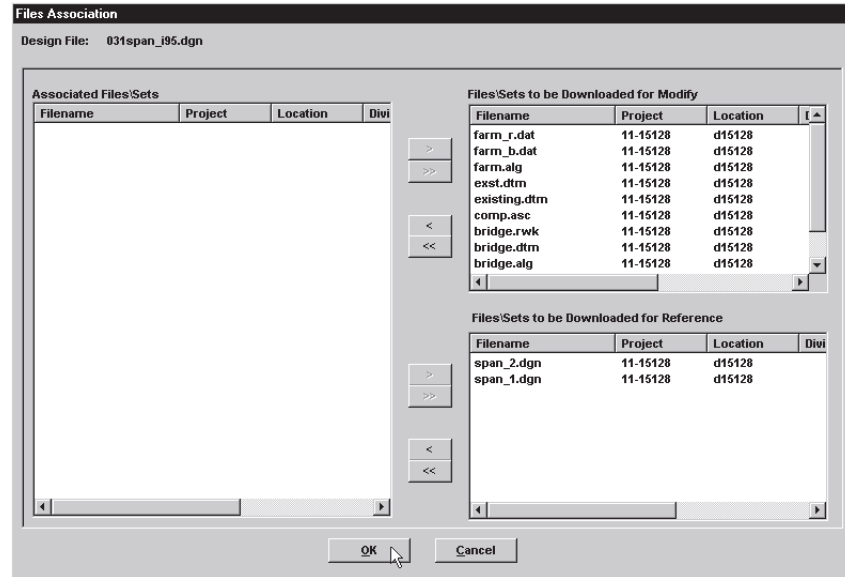
4. In this case, the user chooses to check out all the associated files. (Files selected for 'reference' would be copied to the designated reference directory; their Availability would remain 'On Server'. (For information on the 'reference' directory, see Additional 'igds' Parameters.)

    The document now open in InRoads as well as the files chosen for 'Modify' on the File Association dialog box are checked out to the user's workstation.

To check in a document with new associations:

Let's now look at how File Association handles new files generated during a design session for the 'parent' document opened in the previous exampe. As explained earlier, when the original document is checked in at the end of a session, Falcon/DMS provides a list of any new files generated by the application extension and allows the user to choose which new files are to be added to the database and moved to the data repository, and which are to be discarded.

1. From the File list window, the user selects the design file to be checked in. In the example the contents of the file were modified with InRoads. The modifications produced a several new files—three that will become part of this 'parent' document's dataset and one that will be discarded.

2. The user presses Check In. If no new files were generated during the just completed design session, Falcon/DMS moves the 'parent' document and the associated files back to the data repository, and updates the database without further user intervention.

3. The Add Files to be Associated dialog box opens; the user chooses the files he wishes to add to the database and which files, if any, he wants to delete. The user then presses Continue.
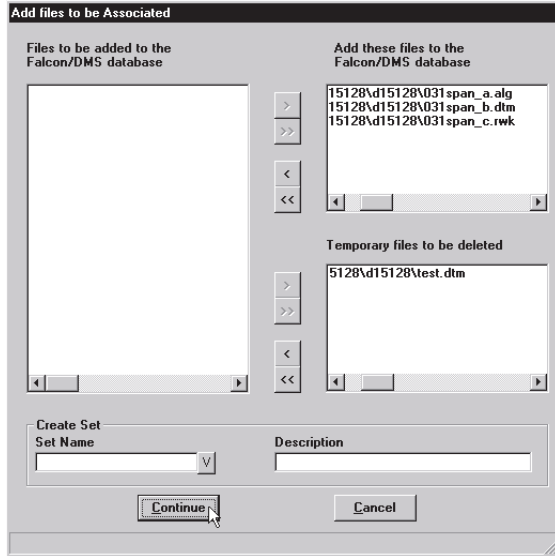
In this example, InRoads generated four new files during the design session. Because File Association was configured to track for the extensions that InRoads assigned to the new files, Falcon/DMS opens the Add Files to be Associated dialog box. When files are added to the database from this form, Falcon/DMS also adds the files to the 'parent' document's list of associated files.

The user can also group files as a set that File Association recognizes the next time the user opens the 'parent' document. File Association allows you to either create a new set or add files to an existing set, provided that set is located in the 'parent' document's home directory.

If the user does not wish to add any documents at this time, but does not want to discard the new files, he simply presses Continue without listing files in either the Add or Delete windows.

4. The Add Associated File dialog box opens. The user assigns a file format and enters description information as required.
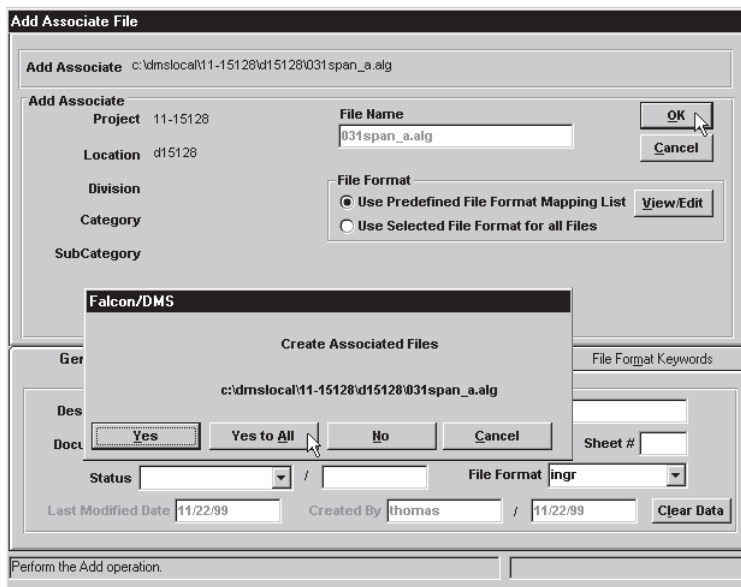
As with the Add Local File and Add Server File commands, the user can assign a file format for each file or use the pre defined File Format Mapping List to automate the process. (When File Associations schemes are properly planned and configured, the File Format Mapping list for the relevant File Formats will have been defined so that user intervention is reduced to a minimum.)

Note: If the design file just checked in did not have any associations, as described in the first example—To open a document with existing associations, File Association would still recognize the new documents indicated in Step 3 and give the user the opporutunity to define associations for these documents.
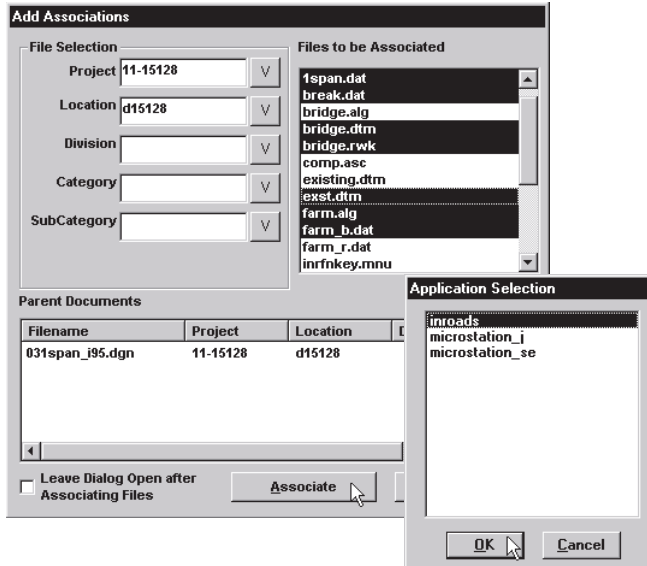
To define a File Association scheme:

The scenarios that we have considered in the last two examples explain how to manage documents for pre defined File Association schemes. Let's now consider how a File Association scheme is initially defined for documents already on the data repository and under the control of Falcon/DMS.

1. From the File list window, the user selects the design file for which he will establish associations. If an association scheme for multiple files is required, list the two or more files in the Files Selected window.

2. From the Document Manager's menu bar, choose File > File Association > Add to…
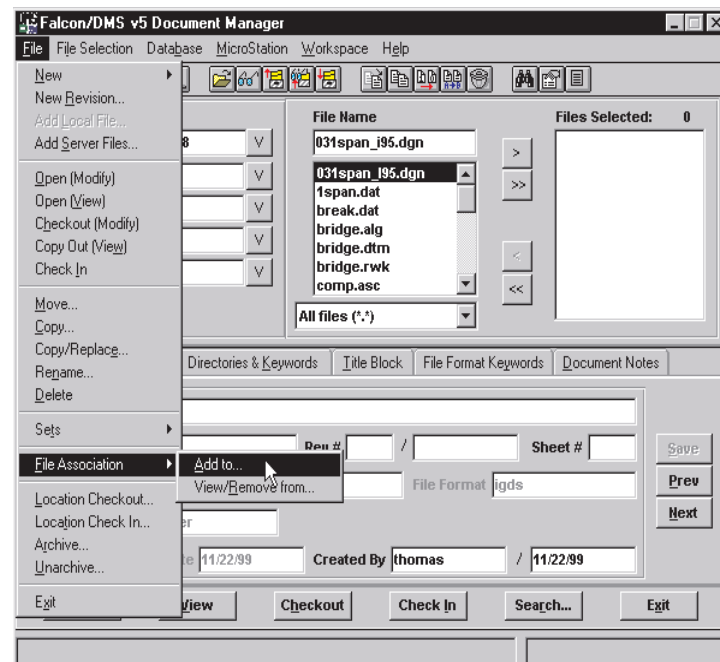
3. On the Add Associations dialog box, the user selects the file or files to be associated; for multiple files use the standard Windows <Shift> or <Ctrl> conventions. Note that File Association allows you to select files from multiple directories.

4. Press Associate. The Application Selection dialog box opens; the user chooses the application for which the selected files will be associated.

Add Associations

File Selection

Project 11-15128 ∨
Location d15128 ∨
Division ∨
Category ∨
SubCategory ∨

Files to be Associated

1span.dat
break.dat
bridge.alg
bridge.dtm
bridge.rwk
comp.asc
existing.dtm
exst.dtm
farm.alg
farm_b.dat
farm_r.dat
inrfnkey.mnu

Parent Documents

Filename	Project	Location	D
031span_i95.dgn	11-15128	d15128	

☐ Leave Dialog Open after Associating Files [Associate]

Application Selection

inroads
microstation_i
microstation_se

[OK] [Cancel]

Note: As you will see, special attention is paid to defining the association rules for a particular application. In order for an application to be listed in the Application Selection dialog box the Falcon/DMS administrator must insure that two things occur:

❏ At least one filename or wildcard definition must be defined for that application in the File Formats & Applications environment.

❏ The application must be assigned to the File Format of the 'parent' document candidate.

These issues will be explained in detail in the section on defining File Association parameters.

Configuring File Association

Attention: In order for users to see the File Association commands on the Document Manager and Falcon/DMS to recognize existing associations, File Association must be enabled on the Sytem Wide Toggles dialog box.

File Association is defined from the File Formats & Applications environment and is accessed from the Applications Properties tab.

File Formats / Applications (Global Tables)

File Formats	Application Properties	File Format Mapping

Application	DDE	Command Line	
acrobat	No	c:\acroread\acrord32.exe	[Save]
autocad	Yes	c:\program files\autocad r14\acad.exe	[Exit]
excel	Yes	c:\program files\microsoft office\office\excel.exe	
indesign	No	c:\program files\adobe\indesign\indesign.exe	[Edit]
inroads	Yes	c:\win32app\ingr\inroads\bin\civil.exe	[Add]
microstation_j	Yes	d:\bentley\ustnj\program\microstation\ustation.exe	[Remove]

[Launch File Association] [Help]

Edit
Application
[   ]
Command Line
[   ] [Browse...] [OK] [Cancel Edit]
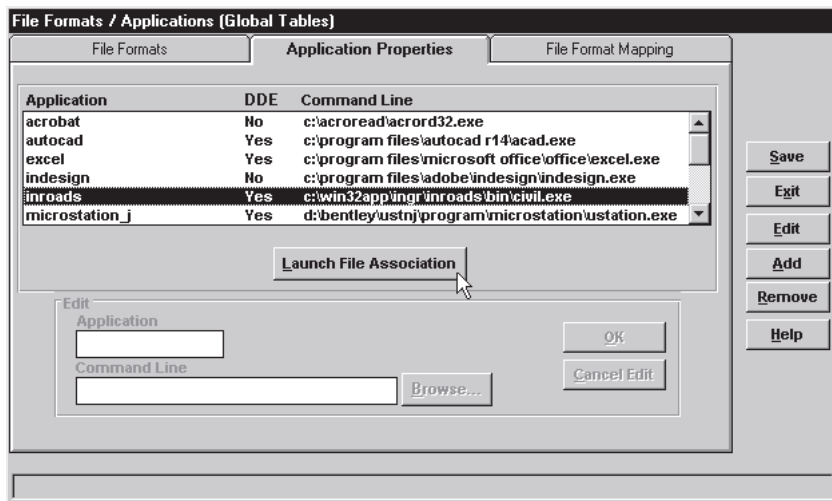
To open the File Association dialog box, select an application and press Launch File Association. This form is used to add, edit, activate, and remove File Association specifications for the current application.
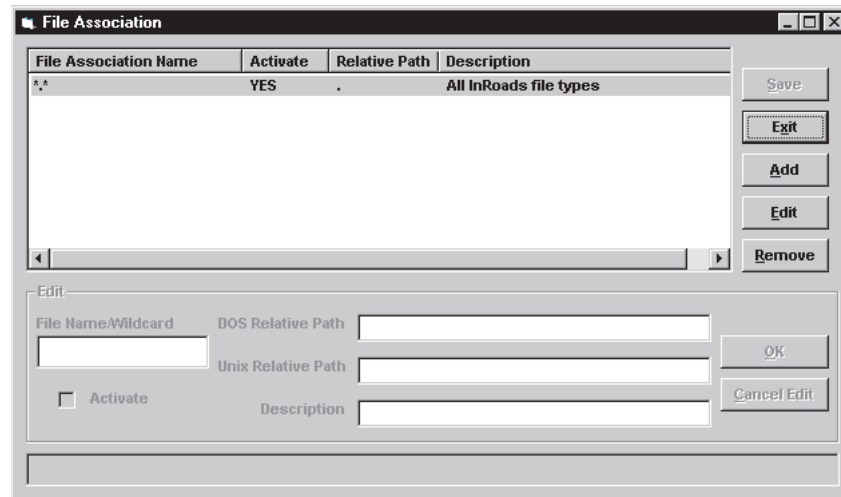
A File Association specification includes a specific filename or a wildcard, an activation state, a relative path (for either DOS or Unix), and a description:

❏ **Filename/Wildcard [option 1]:** In many cases a '*.*' wildcard enables File Association to account for any filename or wildcard combination.
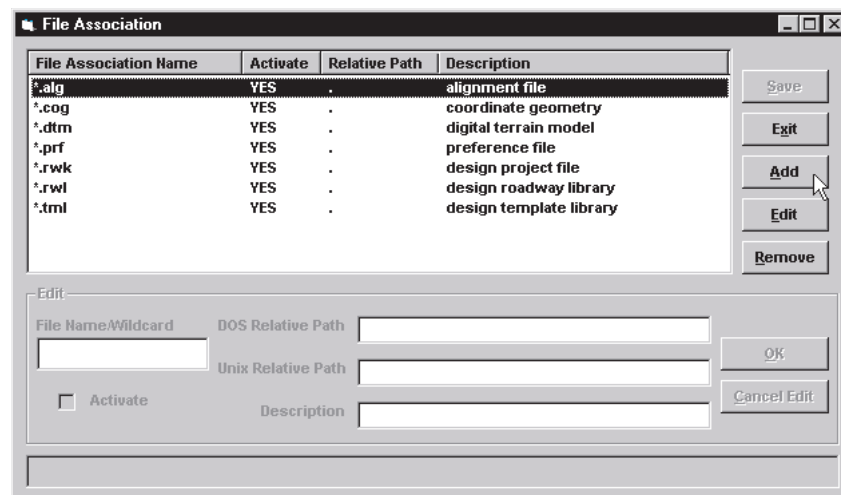
❏ **Filename/Wildcard [option 2]:** Define specific filenames or wildcards (i.e. *.dtm, *.rlk, *.alg, etc) to filter the list of files that displays when File Association is used to automatically add new files to the database and the 'parent' document's list of associated files as the 'parent' document is checked in.

❏ **Activate:** Check 'Activate' to enable the current Filename/Wildcard.

❏ **Relative Path:** This setting determines where Falcon/DMS places associated 'child' files that are checked out with a 'parent' document, relative to the current application's local working directory. In the example above the relative path is set using a period; this causes Falcon/DMS to deliver the associated 'child' files to the same directory as that of the parent document.

❏ **Description:** A description for each filename/wildcard items is required (40 character maximimun).

The File Association table

To add an association:

1. Select the Applications Properties tab.

2. Select an application.

3. Press Launch File Association.

4. On the File Association dialog box, press Add.

5. Enter a filename or wildcard. The filename length is determined by Maximum Fielaname Length setting in System Wide Toggles (the maximum is 80 characters).

6. Enable the Activate checkbox.

7. Define a DOS relative path: use a period to (.) to direct Falcon/DMS to place 'child' files in the same directory as the 'parent' document. Specify a directory name to build a destination directory relative to the application's working directory (typcially, c:\dmslocal). If required, define a Unix relative directory using the rules.

8. Enter a description.

9. Press OK. Repeat as required.

10. Press Cancel Add, Save, and Exit.

To edit an association:

1. In the File Association Name list, double-click an entry, or select an entry and press Edit.

2. Edit as required.

3. Press OK. Repeat as needed.

4. Press Cancel Edit and Save.

To remove an association

1. In the File Association Name list, select an entry.

2. Press Remove. Repeat as needed.

3. Press Save.

The Final Steps

File Formats tab: Always add any application for which you have defined File Association parameters to the relevant File Format's application list; this list is maintained on the File Formats tab under 'Applications'. For example, 'igds', the MicroStation File Format, would list MicroStation J as the application which automatically launches when a user opens a document with a File Format of 'igds'. In order for the Falcon/DMS to now recognize associations or potential associations for InRoads, the user user must be able to launch the InRoads flavor of MicroStation.

System Wide Toggles: Enable File Association on the Sytem Wide Toggles dialog box.

The File Format Mapping tab

The File Format Mapping tab is used to add, edit, and remove File Format > Extension mappings used by the Add Local, Add Server, and Check In commands.
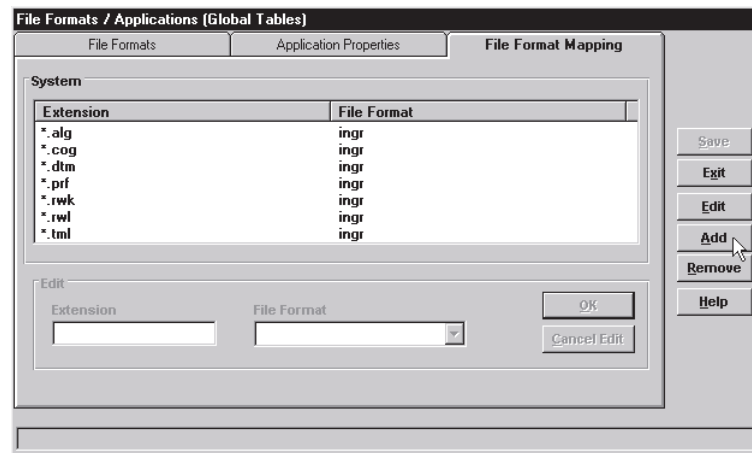
Falcon/DMS has a number of Document Manager commands designed to create database records. Add Local File and Add Server File are provided expressly for the purpose of adding files to the database. Another, the Check In command, enables files that are generated in a local working directory during a design or work session to be moved to the data repository, and as part of that operation, Falcon/DMS creates database records for the files.

This feature of the Check In command was designed to make the process of adding files that are generated by an application, doing precisely what it is designed to do, as transparent to the user as possible. Before explaining how the File Format Mapping feature facilitates this, let's first review the steps that occur when a user attempts to check in a group of documents that contains files that do not yet have database records.

What the user sees
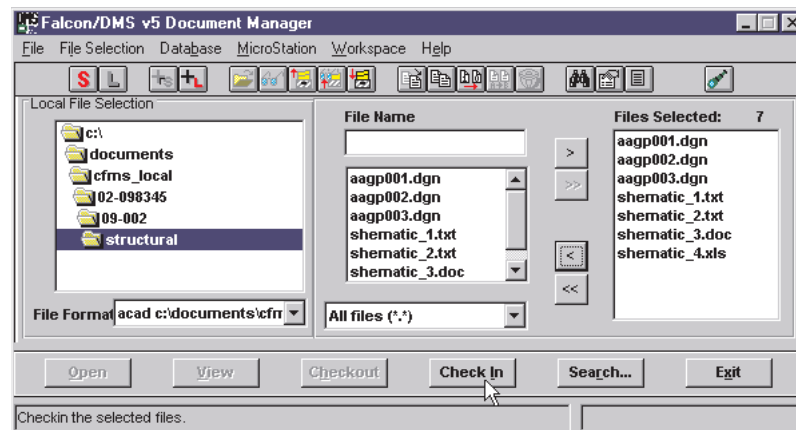
1. Set File Selection to Local File Selection.

2. Navigate to the local directory containing the documents to be checked in.

3. Select both the documents that already have database records and the documents to be added to the database as part of the Check In operation.

4. Press Check In.

5. Falcon/DMS first processes the documents with database records, moving them to the data repository and updating their database records.

6. Falcon/DMS next displays a message indicating that the remaining files are not in the database and asks if you wish to create records for them. Press Yes.
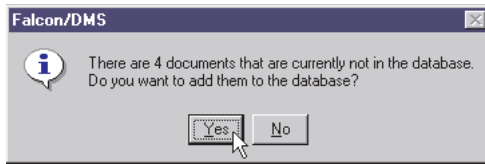
7. The Add Local Document dialog box opens.
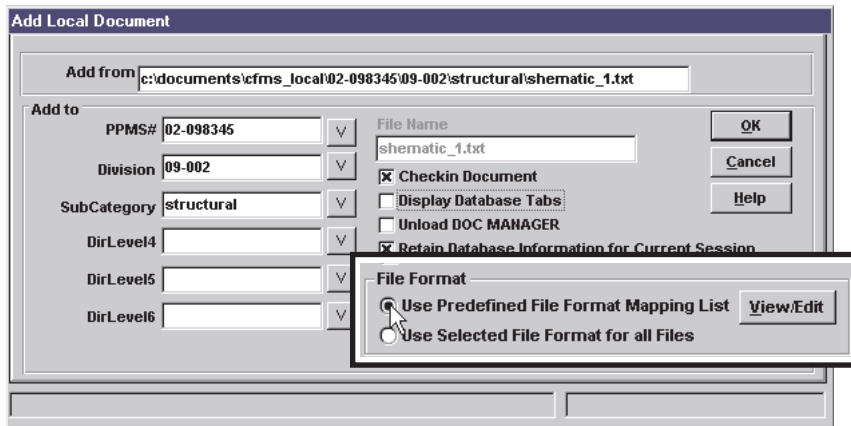
In the Add To > File Format region, the default setting is for the Predefined File Format Mapping List.

Press OK. One of two things will happen—the user answers Yes To All and the transaction completes, or the user manually specifies the File Formats.

When the File Format Mapping table contains a mapping for each file extension that occurs in the list of files that do not have database records, these files will be automatically added to the database and moved to the data repository when the user presses OK and Yes To All.

For any file with an extension that is not listed in the File Format Mapping table, the user is required to designate a File Format before Falcon/DMS moves each file to the data repository and adds it to the database.

Note: The user is able to build a local File Format Mapping table that contains extension to File Format associations not listed in the global table.

The File Format Mapping table

To add a File Format Mapping entry:

1. Select the File Format Mapping tab.

2. Press Add.

3. In the Extension input field, key in an extension

4. From the File Format drop–down list box, select a File Format.

5. Press OK. Repeat Steps 3 & 4 as needed.

6. Press Cancel Add and Save.

To edit a File Format Mapping entry:

1. In the Extension/File Format list, double-click a mapping, or select a mapping and press Edit.

2. Edit the Extension and File Format fields as required.

3. Press OK. Repeat as needed.
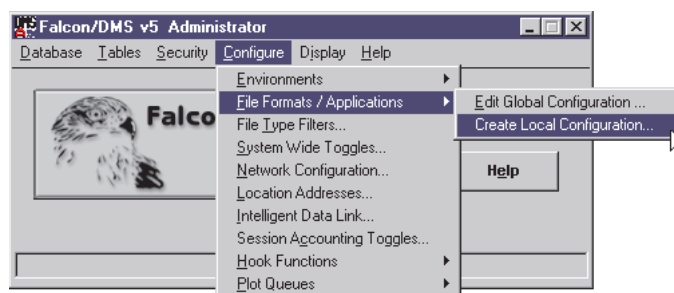
4. Press Cancel Edit and Save.

To remove a File Format Mapping entry:

1. In the Extension/File Format list, select a mapping.

2. Press Remove. Repeat as needed.

3. Press Save.

Global vs Local File Format Configuration

The File Formats dialog box assigns its settings to one of two configuration categories: Global or Local. While the ability to override the Global Configuration with a set of 'local' configuration files remains, with the introduction of Multiple Environments it is no longer the most efficient way to assign special configurations to individual users (see Chapter 7). If you choose not to use Multiple Environments to control File Format & Application configurations the Global vs Local File Format configuration option continues to be a workable approach.

The Global Configuration maintains file format and application settings in configuration files on the application server that are read by any workstation that does not have a Local Configuration. The files, APPLCMD.CFG and FILEFMT.CFG, are located in your application server's <falcon_sdir>\cfg directory. When versions of these files are present in a workstation's <falcon_dir>\cfg directory, Falcon/DMS reads these files rather than the global settings

There are several ways to define a Local Configuration:

❏ From the administrator's workstation, open the Falcon/DMS Administrator. Select Configure > File Formats/Applications > Create Local Configuration—the configuration files are written to the <falcon_sdir>\cfg directory on the administrator's workstation. Finally, move these files from the administrator's workstation to the target workstation.

❏ From the user's workstation, open the Falcon/DMS Administrator. Select Configure > File Formats/Applications > Create Local Configuration—the configuration files are written to the <falcon_sdir>\cfg directory on the user's workstation.

To reconnect a user to the Global Configuration, delete the local configuration files

(APPLCMD.CFG and FILEFMT.CFG) from the user's workstation. This can be done manu-
ally or from the Falcon/DMS Administrator—when local files are present a Delete Local
Configuration command is available on the Configure > File Format/ Applications menu.
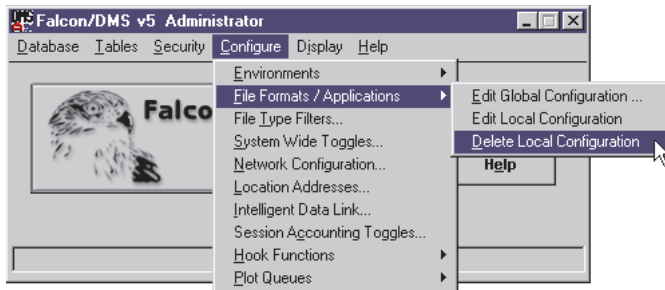
Controling Non-Electronic Documents

The Non–electronic Document Control feature permits you to maintain records for non–
electronic documents in the Falcon/DMS database.

The database record for a non–electronic document is not associated with a file maintained
on a computer. Instead, for example, the database record for a non–electronic document
may list information about a piece of paper in a file drawer. This information could include
a drawing number, a description, and the cabinet and drawer number of the physical
document. If a person removes the original document from a file room, the administrator
of that facility can edit that document's database record to identify who removed it, when it
was removed, where it was taken, and when it is to be returned.

## Non-Electronic Document Control Issues

The only requirement for non-electronic document control is to define a file format with
the 'electronic' setting disabled.

However, to properly prepare your Falcon/DMS installation to effectively manage non-
electronic documents does require some additional planning. Consider the following
suggestions:

❏ Setup a directory structure that matches the physical organization of your paper
   document cabinets, plan files, or drawers. Though these directories remain empty,
   the structure itself facilitates database searches.

❏ Setup an environment specifically for non-electronic document control. Such an
   environment will keep the empty directory structure separated from the projects
   that contain electronic documents.

❏ As part of the non-electronic document environment, use keywords to assist in the
   tracking of paper documents that are removed from your facility (see Chapter 9).
   For example, this version of Falcon/DMS identifies the person logged on to the
   workstation as the individual to whom a document is checked out—a more accurate
   document record could use a keyword to identify that person.

Configuring Falcon/DMS for Non-Electronic Document

The only actual setup for non-electronic document control is to define a file format with the 'electronic' setting disabled.
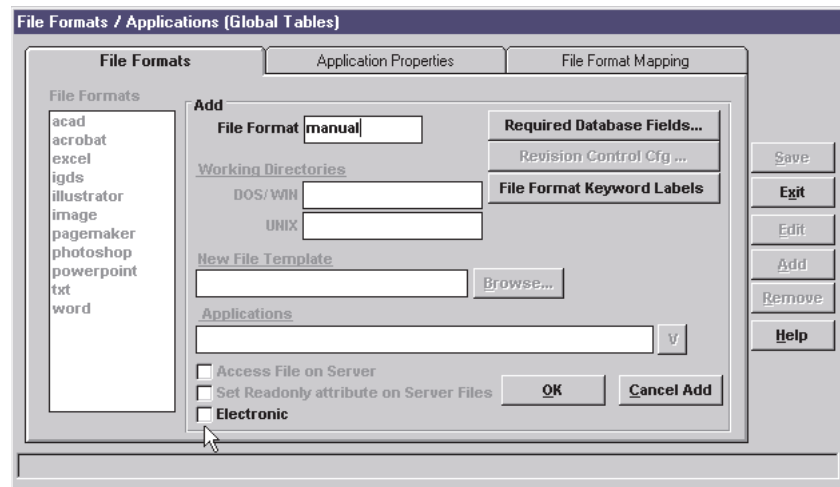
To define a non-electronic File Format:

1. From the Administrator, select Configure > File Formats/Applications. The File Formats/Applications dialog box opens.

2. Press Add. Enter a File Format name and uncheck the Electronic check box.

    All options except Required Database Fields and File Format Keyword Labels are disabled.

3. Press OK, Cancel Add, and Save.

Managing Non-Electronic Document

The Document Manager commands that are valid for non-electronic documents are Checkout, Check In, Move, Copy, Rename, and Delete. Because a non-electronic document in Falcon/DMS is represented by only a database record, certain Document Manager commands are meaningless—the Open, View, Checkout for View, Add Server, and Add Local File commands all require that an actual file exist. Finally, the database records for non-electronic documents are visible in the Document Manager's file list windows only when the Server View Mode user preference is set to Database. When the Server View Mode user preference is set to Directory, non-electronic documents are visible only from a search result.

Falcon/DMS

The Administrator

Chapter 4:
MassAdd & The
Master 'Project'
Table
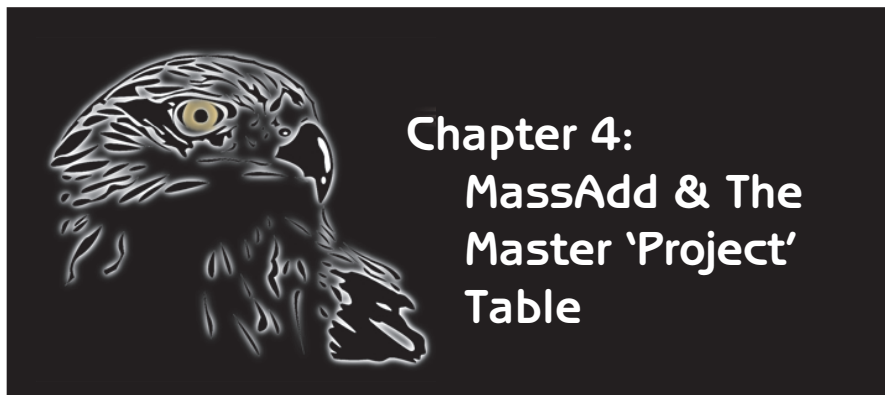
The focus of this chapter is on the Mass Add utility and the Master 'Project' Table.

MassAdd is used to 'bulk load' the Master 'Project' Table with directory names, and the database with new records. That is, to create database records for existing, or 'legacy', documents.

The Master 'Project' Table appears in both the Administrator and the Document Manager (Windows and MDL). In the Document Manager, the Master 'Project' Table provides access to the project directories over which Falcon/DMS has control. In the Administrator, the Master 'Project' Table is used to maintain this list of project directories.

(The term 'Project', as it applies to the Master 'Project' Table is variable. You may use any label you wish to describe a master, or top-level, directory under which a project's data structure and files are found. See Chapter 5.)

Once the Master 'Project' Table is built, new projects can be added from either MassAdd or from the Master 'Project' Table dialog box.

Because a typical installation and implementation workflow requires that a Master 'Project' Table be built before database records can be created, we will begin our discussion with MassAdd and its Add Projects command.

MassAdd

The MassAdd utility serves two purposes:

❏ To define the Master 'Project' Table. This table lists the top-level directories on the data repository, the server containing the documents Falcon/DMS will control. The Master 'Project' Table is used by the Windows and MDL Document Managers to give users access to specific project directories on data repositories.

❏ To bulk load the database. Bulk loading the database means that a single operation you create database records for files located in one, or several, or all project directories, while associating each file's record with the appropriate File Format.

To open MassAdd:

From the Start button, select Programs > Falcon_DMS > MassAdd or, in the Falcon_DMS program folder, double-click the MassAdd icon.

To define the Master 'Project' Table:

1. On the MassAdd form, press Add Projects... The Add Project dialog box opens (opposite page).
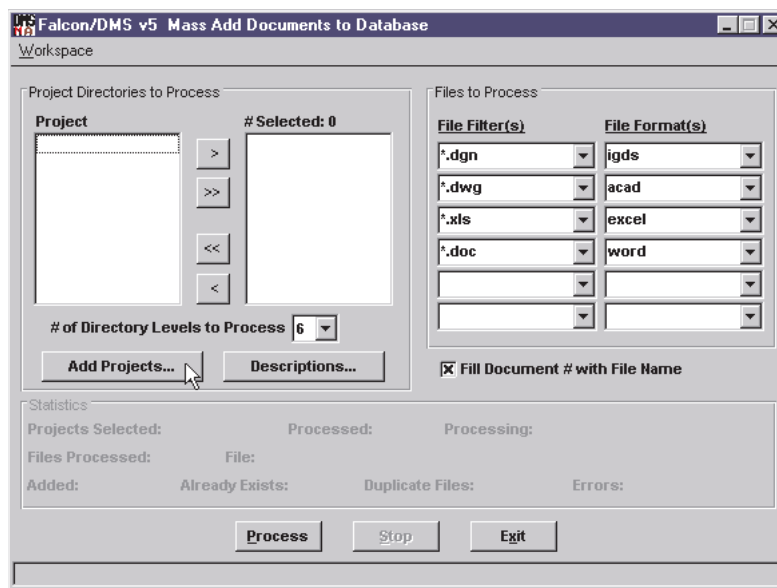
2. From the Server drop-down list box, select a data repository.

   From the Resource drop-down list box, select the resource from which MassAdd will build a project list.

3. Press Build Project List. All the directories available under the selected resource display in the Projects Not Added list.

4. From the Projects Not Added list, select the projects you wish to add to the Master 'Project' List.

5. Press Process. When completed, press Exit.

   Repeat this process for different combinations of servers and resources, as needed.

Bulk loading the database

MassAdd assigns a File Format to each document as it is added to the database. In the File to Process region, you will define a set of up to six 'Extension (File Filter)—File Format' combinations. MassAdd scans each project directory structure for files that match these combinations. Ignoring documents with extensions not listed, MassAdd creates a database record for each document with an extension indicated in the Files to Process list.
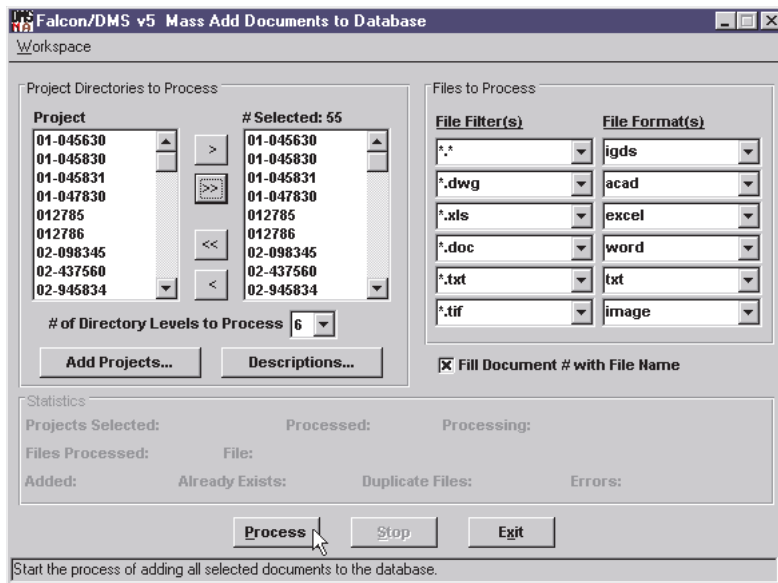
In the example above, MassAdd requires only one pass of the selected projects. The specified extensions, or File Filters, cause files with those extensions to be added with the indicated File Formats assigned. All other documents within the selected projects that do not conform to the five specified extensions will be added to the database with the 'igds', or MicroStation, File Format.

To bulk load the database:

1. In the Project Directories to Process region, select the projects that MassAdd is to scan.

2. Also in the Files to Process region, use the Directory Levels to Process drop-down list box to indicated the subdirectory depth to which MassAdd is to scan for files.

3. In the Files to Process region, define the necessary File Filter—File Format combinations.

4. Press Process.

   MassAdd generates a log file that lists all the projects that were processed, a breakdown of each project's subdirectory structure, the number files added from each subdirectory, and the number of duplicate files occurring within each project.

   The summary includes a breakdown duplicate files by location and filename. The summary also identifies the file that is already in the database and its location.

   How Falcon/DMS identifies a file as a duplicate is determined by the Primary Index Key. That is, when the Primary Index Key is set to <Filename>, any subsequent use of a given filename that matches the search criteria anywhere on the data repository, will cause MassAdd to flag the occurrence as a duplicate file. When the index key is set to <DirLevel1,Filename>, any subsequent use of a given filename that matches the search criteria anywhere within a project's directory structure, will flag the occurrence as a duplicate file, etc.

To add project descriptions

When they have been defined, Project Descriptions, can be displayed as a user designates a current project on the Document Manager. The user controls this display by toggling the Project Descriptions user preference.

As you will see in the following section on the Administrator's Master 'Project' Table, Falcon/DMS allows you to add a description as you create each project, or add a description while editing a single project. MassAdd also allows you to add a description to each project. However, consistent with MassAdd's 'bulk load' approach to operations, the Add Project Descriptions allows you to provide project descriptions to multiple projects in a one operation. The Add Project Descriptions command imports a comma delimited file that must contain two columns separated by a comma—the first column lists the projects, the second column lists each project's description. After creating the description file, use the following procedure to update the Master 'Project' Table.

1. On the MassAdd dialog box, press Description. The Add Project Descriptions dialog box opens.

2. Enter the directory and filename of the comma delimited description file. Press Process.
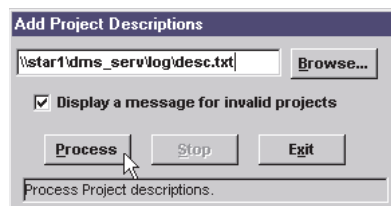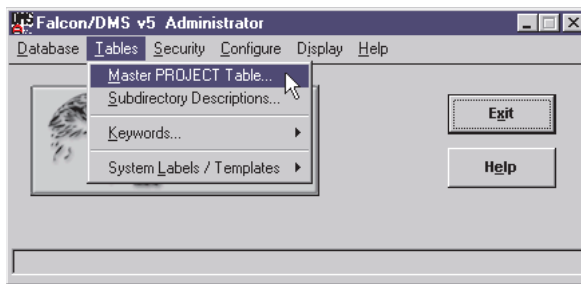
The Master 'Project' Table

Note: A number of dialog boxes use Data System Labels (see Chapter 5) to assign the name that appears on the form's control bar, as well as the name of the menu option used to open that form. When the Master 'Project' Table dialog box opens, the name that appears in its control bar will display the Directory Level 1 label defined in the Data System Labels environment. For example, the first Tables pull-down menu item pictured below is the Master PROJECT Table command; the control bar on the associated dialog box identifies the form as the Master Project Table. If you were to change the Directory Level 1 in the Data System Labels environment from 'Project' to 'Client', 'Client' would replace 'Project' wherever 'Project' occurs in the application interfaces.

Use the Master 'Project' Table to:

❏ Add and edit projects.
❏ Remove projects and/or their database records.
❏ Create actual project directories on the data repository [optional].
❏ Create actual project subdirectories on the data repository [optional].
❏ Assign the Server/Resource combination for each project.
❏ Activate or deactivate projects.

To add a project:

1. From the Administrator, select Tables > Master 'PROJECT' Table.

2. On the Master 'Project' Table dialog box, press Add.

3. In the Description field, key in a description [optional].

4. From the Status drop-down list box, change the Status if necessary.

    Inactive suppresses the display of a project in the Document Manager's project list.

5. Select the Server and Resource—if only one server and/or resource has been defined, the list boxes show that single entry as the default.
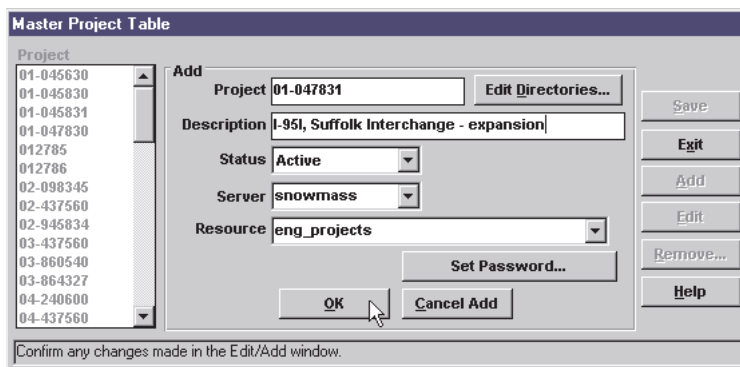
6. Press OK.

☞ ATTENTION

FTP CLIENT/SERVER NETWORK CONFIGURATION ONLY—While you are able to create a project entry in the Master 'Project' Table, due to a limitation of native FTP you will not be able to create the actual project directories nor their subdirectories from the Master 'Project' Table dialog box. There is, however, an effective 'work around'.

The best way to facilitate the creation of actual directories through the Falcon/DMS Administrator, without affecting the global FTP Client/Server configuration, is to define an Environment in which the Network Configuration is set to UNC. (See Chapter 7.) When you wish to use the Master 'Project' Table to add a project AND create the data repository directories, simply set the environment to that configured for UNC.

Be mindful that if your FTP configuration in Falcon/DMS specifies the special account, any directory created on the data repository has to be owned by this account. Therefore, when you switch to the environment that uses UNC, your current log in account for the NT domain must have the same account name as the special account defined for the FTP configuration.
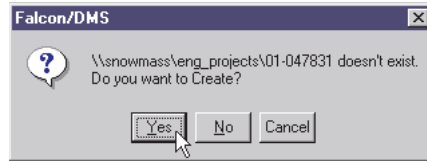
6. If the project directory does not exist on the data repository, Falcon/DMS will ask to create the directory:

❏ To add the project to the table but not the actual directory on the data repository, press No.
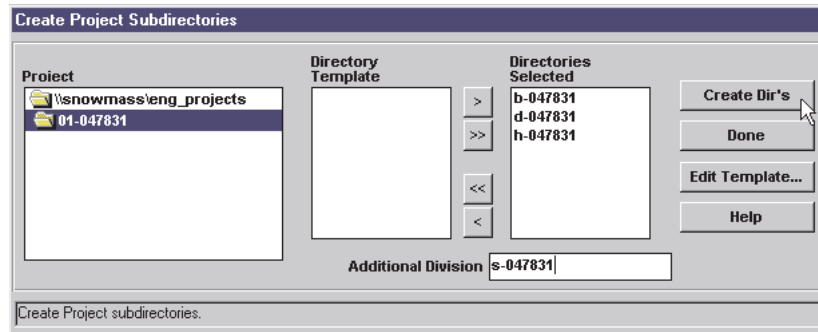
❏ To add the project to the table and to create the actual directory on the data repository, press Yes. The Create Project Subdirectories dialog box opens. To create subdirectories under the new project:
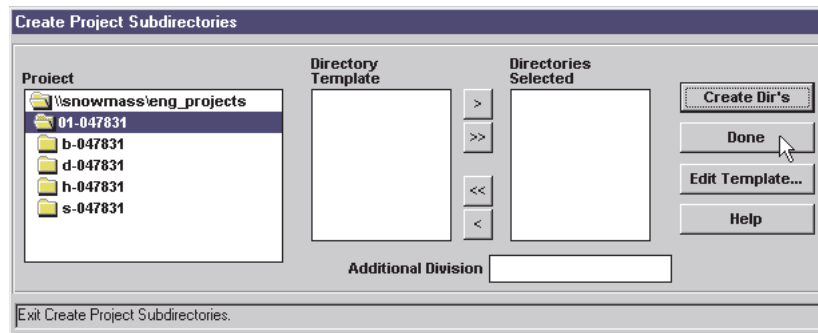

a. In the Additional 'Division' input field, key in a subdirectory name.

b. Press <Enter>. Repeat as needed.

c. Press Create Dir's.

d. Press Done. The Create Project Subdirectories dialog box closes.

7. Repeat Steps 2 – 6 as needed.

8. Press Cancel Add.

9. Press Save and Exit.

Directory Templates:

As you have just seen, the Create Project Subdirectories allows you to create subdirectories under project directories on a data repository without having to rely on operating system commands (e.g. md or mkdir).

For certain data systems it is possible that part or all of the same directory structure occurs within multiple projects—Falcon/DMS provides a way to save these subdirectory schemes in templates. The template is then used to assist in the creation of subdirectories for new projects as they are added. When directory templates exist, the subdirectory names appear in the Directory Template window on the Create Project Subdirectories dialog box. From this window you select the items for which you will create actual directories on the data repository.
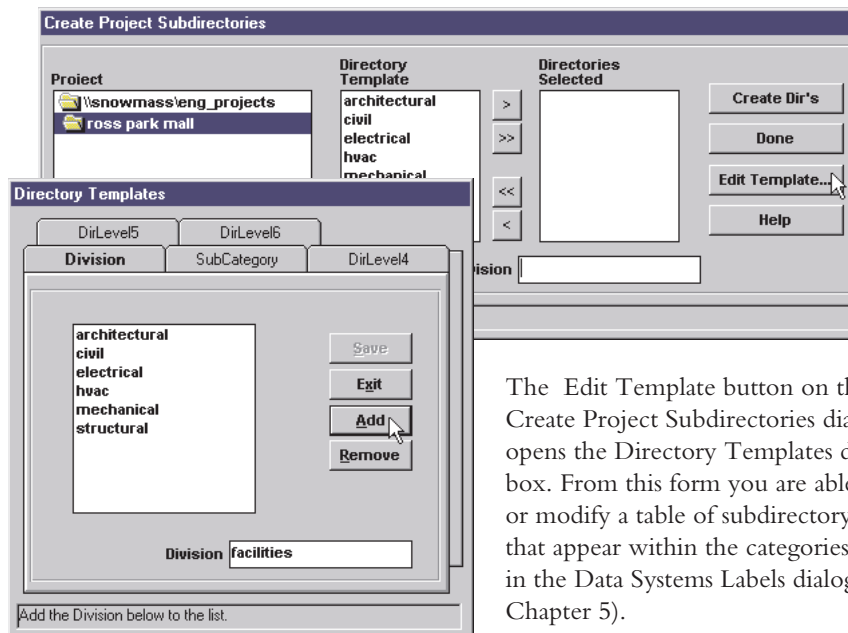
The Edit Template button on the Create Project Subdirectories dialog box opens the Directory Templates dialog box. From this form you are able to build or modify a table of subdirectory names that appear within the categories defined in the Data Systems Labels dialog box (see Chapter 5).

You can add to or otherwise modify any template directory from the Create Project Subdirectories dialog box.

To add to or modify Directory Templates:

1. On the Create Project Subdirectories dialog box, press Edit Template. The Directory Templates dialog box opens.

2. Select a subdirectory level tab—the tab names are Data System Labels and are explained in Chapter 5. In the input field, enter a directory name. Press Add or <Enter>. Repeat as needed.

3. Press Save and Exit.

To edit a project:

1. From the 'Project' list window, double-click on a list item, or select an item and press Edit.

2. Modify as needed. Click OK.

3. Repeat as needed.

4. Press Cancel Edit and Save.

To remove a project:

1. From the 'Project' list window, select a list item.

2. Press Remove. The Remove Project dialog box opens. Select the appropriate option.

❏ 'Project' Entry & Database Records

❏ 'Project' Entry only

❏ 'Project' Database Records only

When you select the first or third option, the All Categories and Specify Categories radio buttons are available. When Specify Categories is selected you are able to choose the subdirectories for which database records are to be deleted.

The actual directories on the data repository are deleted only when they are empty of files.

4. Press Yes or No on the Remove Project dialog box.

5. Press Save and Exit.

Falcon/DMS

The Administrator

Chapter 5:
General
Configuration
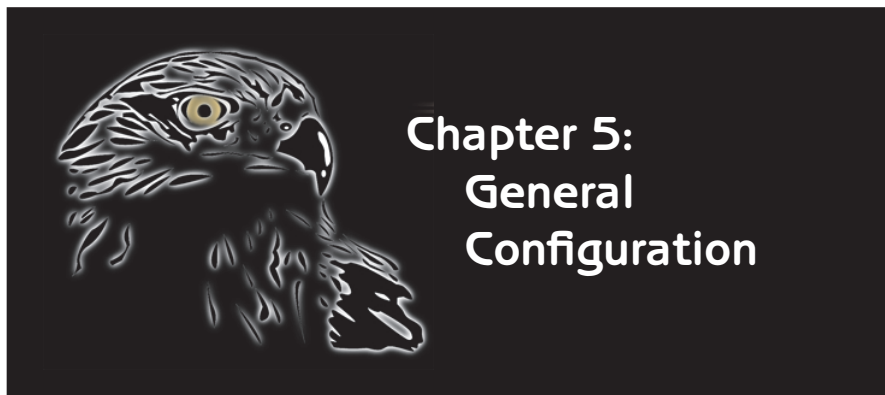
The focus of this chapter is on basic configuration issues. Special emphasis is given to System Wide Toggles environment—this is of particular importance because it is here that the maximum lengths for filenames and directory names are defined. Also defined through System Wide Toggles is . File Format and Applications and File Filters are also covered are the settings required for instant recovery of deleted files, keyword validation for system level keywords, and enabling hook functions and File Association.

Also covered in this chapter is how to customize the data system labels that appear on both versions of the Document Manager, how to create Directory Templates, and how to define both subdirectory descriptions and the contents of the File Filter drop-down list boxes.

System Wide Toggles

The System Wide Toggles dialog box is used to establish a number of important parameters that affect the most fundamental ways in which Falcon/DMS operates. Among the most important of these are the following :

Support for long filenames and directory names: Falcon/DMS supports a maximum filename length of eighty (80) characters and a maximum directory name length of forty (40) characters. It is important to understand that each component, or subdirectory, of the full directory path specification can contain up to forty characters.
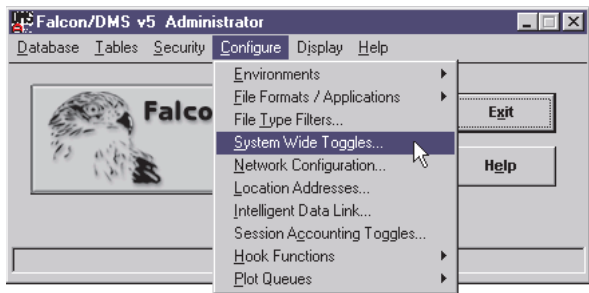
Instant recovery for deleted files: Falcon/DMS is usually configured to move files to a secure directory on the network when the Document Manager is used to delete documents. Rather than deleting the actual files, Falcon/DMS gives that responsibility to the administrator, who removes the files after sufficient time has elapsed to insure that the documents are, indeed, no longer needed.

Keyword validation: When adherence to standards is required for assigning descriptive database attributes, users able to assign only those keyword appearing in predefined lists. If a user chooses to key in a keyword, that item must appear in the associated list box.
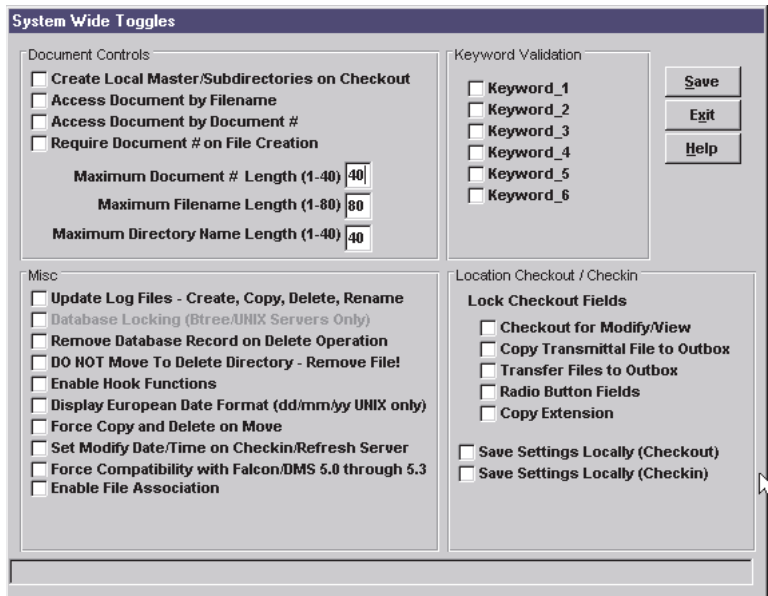
Enabling of Hook Functions and File Association: The Hook Functions control and the File Association control are located in System Wide Toggles.

To open the System Wide Toggles dialog box

From the Administrator, choose Configure > System Wide Toggles.

The System Wide Toggles dialog box

| GROUP	CONTROL
Document Controls	Create Local Master/Subdirectories on Checkout
	When enabled, Falcon/DMS replicates a document's server-based home directory path under the document's local working directory as it is checked out or opened. A local path is removed when the last file is checked in. This option is automatically enabled for any database with a Primary Index Key of DirLevel1;Filename or higher.
	Access Document by Filename
	Automatically enabled. Permits users to query the database and manipulate a document by its filename. This option may be disabled only when Access Document by Document # is enabled.
	Access Document by Document #:
	This option permits users to query the database and manipulate a document by its document #. This option may be disabled only when Access Documents by Filename is enabled.
	Require Document # on File Creation:
	When enabled, this option requires users to define a document # as a file is created.
	Maximum Document # Length (1-40) (Maximum - SQL: 40 / Btree: 14)
	This field specifies the maximum number of characters permitted for a record's 'document #'. The maximum 'document #' length is 40 characters.
	Maximum Filename Length (1-80) (Maximum - SQL: 80 / Btree: 14)
	This field specifies the maximum number of characters permitted for a document's filename. The maximum filename length is 80 characters.
	Maximum Directory Name Length (1-40) (Maximum - SQL: 40 / Btree: 8)
	This field specifies the maximum number of characters permitted for each directory name in a directory path. The maximum length for each component is 40 characters.
Misc.	Update Log Files - Create, Copy, Delete, Rename:
	Causes log files for the Create, Copy, Delete, and Rename commands to be updated.
	Database Record Locking (B-Tree/UNIX Servers Only):
	Replaces the Unix LOCKD program—for B-Tree database installations only.
	Remove Database Record On Delete Operation:
	A deleted document's record is removed from the database. When not enabled, a deleted document's record is indicated as 'Marked For Deletion'. The database must be purged in order for such records to be removed.
	DO NOT Move To Delete Directory - Remove File:
	Documents are removed as users delete them. When not enabled (the preferred approach), the deleted file is actually moved to the (m):\delete directory under the Falcon/DMS server directory. Files in the (m):\delete directory must be manually deleted by the administrator.
	Enable Hook Functions:
	Permits custom hook functions to be operational.
	Display European Date Format:
	Displays dates in the dd/mm/yy format.
	Force Copy and Delete on Move:
	Use operating system's copy command instead of move command.
	Set Modify Date/Time on Checkin/Refresh Server:
	Changes Modify Date/Time from time at which a document closes to the time at which it is checked in to the server.

Misc. (cont.)	Force Compatibility with Falcon/DMS v5.0 through v5.3:
	Closes access to features found in Falcon/DMS v5.4 and later.
	Enable File Association
	Permits File Association to be operational..

GROUP	CONTROL
Keyword Validation	When enabled for any or all of the keyword items, users are forced to select keywords from the Falcon/DMS keyword table. When any of these options are not enabled, users may define their own keywords.
Location Checkout/In	When the Location commands are made available to users not in the administration group, you will often want to control what location options are available. The System Wide Toggles dialog box allows you to lock certain fields and options on the Location form to insure that use of the Location commands comply with your standards.
	Lock Checkout Fields:
	Controls access to features on the Location Checkout dialog box.
	Save Settings Locally (Checkout):
	Permits the settings for the Location Checkout dialog box to be saved as the default.
	Save Settings Locally (Checkin):
	Permits the settings for the Location Checkin dialog box to be saved as the default.

Data System Labels & Templates

Defining Directory Categories

Falcon/DMS permits you to organize your documents in a directory structure that can be from one to six levels deep and to define your own labels for each directory level. If your business is a manufacturing concern you may choose to organize your files within 'product' categories, while an engineering company might organize its files by 'project' or 'client'—Falcon/DMS doesn't restrict you.

To develop this file organization structure you must first define labels for your data system—think of these labels as categories of directories. The Data System Labels dialog box provides six fields that correspond to the six-level subdirectory structure supported by Falcon/DMS. It is not necessary that you use all six levels—the example shown below uses only three.
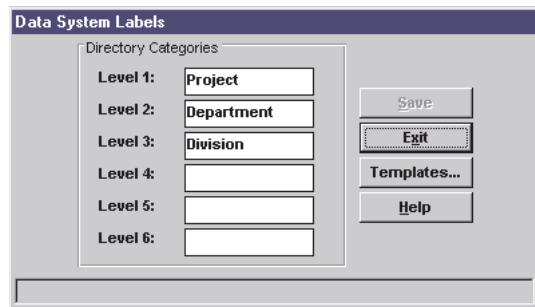
Data System Labels—important features:

❏ When System Labels are defined in the Data System Labels dialog box, directories are not created, they are simply labeled or categorized.

❏ Labels are dynamic and can be changed without altering the underlying directory structure—when a data system label is changed, any dialog boxes that displays Data System Labels will reflect that change.

To open the Data System Labels dialog box:

From the Administrator, choose Tables > System Labels/Templates > Directory Labels/Templates.
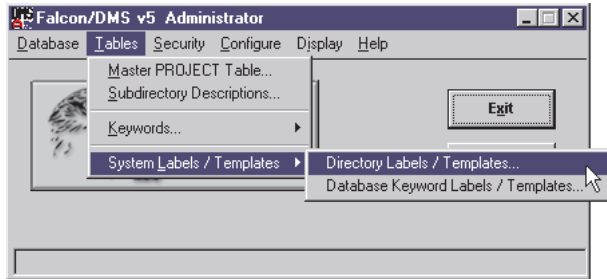

To create labels/categories for directory levels:

1. In the Data System Labels dialog box, enter a label for each level that you wish to display. The fields must be labeled from top to bottom. A field cannot be skipped. Each field has a 12 character limit.

2. Press Save.

What the user sees

The particular Directory System Labels scheme used in the explanation is reflected in the Windows Document Manager shown below. Note that because no label was assigned to directory level 4 through directory level 6, those directory levels do not display on neither the Windows Document Manager nor the MicroStation (MDL) version.
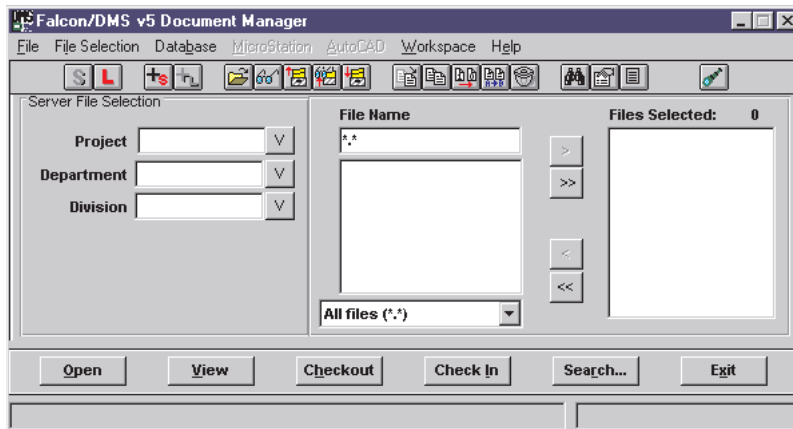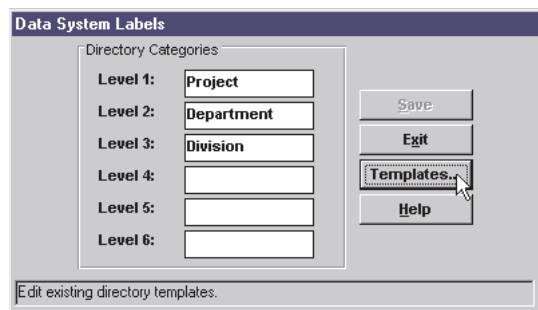
Directory Templates

For certain data systems it is possible that part or all of the same directory structure occurs within multiple projects—Falcon/DMS provides a way to save these subdirectory schemes in templates. The template is then used to assist in the creation of subdirectories for new projects as they are added. When directory templates exist, the subdirectory names appear in the Directory Template window on the Create Project Subdirectories dialog box (see Chapter 4). From this window you select the items for which you will create actual directories on the data repository.

The Edit Template button on the Create Project Subdirectories dialog box opens the Directory Templates dialog box. From this form you are able to build or modify a table of subdirectory names that appear within the categories defined in the Data Systems Labels dialog box (see Chapter 5).

To open the Data System Labels dialog box:

From the Administrator, choose Tables > System Labels/Templates > Directory Labels/Templates.

On the Data System Labels dialog box, press Templates.

To add an item to Directory Templates:

Note that the Level 1 label does not have a tab. The Level 1 label assigns its category/label to the Master 'Project' Table where the level 1 directories are defined.

1. On the Directory Templates dialog box, select a tab.

2. In the input field, enter a directory name. Press Add or <Enter>. Repeat as needed.

3. Press Save and Exit.
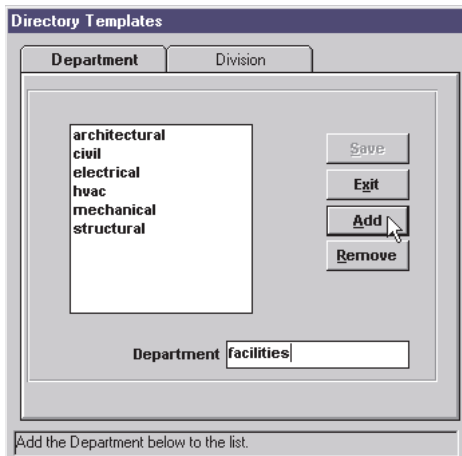
To edit a subdirectory name:

1. On the Directory Templates dialog box, select a tab.

2. In the list window, double-click on a subdirectory name. Edit as required. Press Add or <Enter>. Repeat as needed.

3. Press Save and Exit.

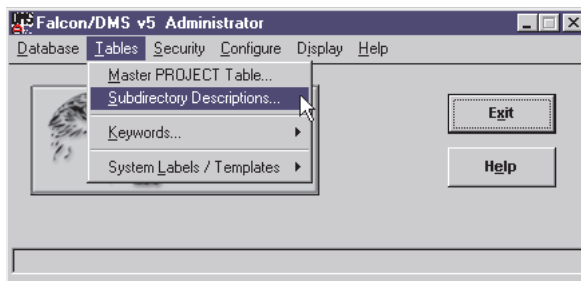To remove a subdirectory name:

1. On the Directory Templates dialog box, select a tab.

2. In the list window, select a subdirectory name. Press Remove. Repeat as needed.

3. Press Save and Exit.

Subdirectory Descriptions

Subdirectory Descriptions are defined from the Administrator and are available as a Document Manager user preference. If the user preference is enabled, when a user select a subdirectory on the Document Manager the list box shows both the directory name *and* a description. Directory Descriptions can be created for each unique directory name occurring within the 5 levels of subdirectories found under the project directories.

To open the Subdirectory Descriptions dialog box:

From the Administrator, choose Tables > Subdirectory Descriptions.

To add a subdirectory description:

1 From the Directory Level drop down list box, select a directory level category.

2 Press Add.

3 In the Directory and Description fields, key in an a directory name and a description, respectively—the Directory field has an 20 character limit; the Description field has a 25 character limit.

4 Press OK and Cancel Add. Repeat as needed

5 Press Save.
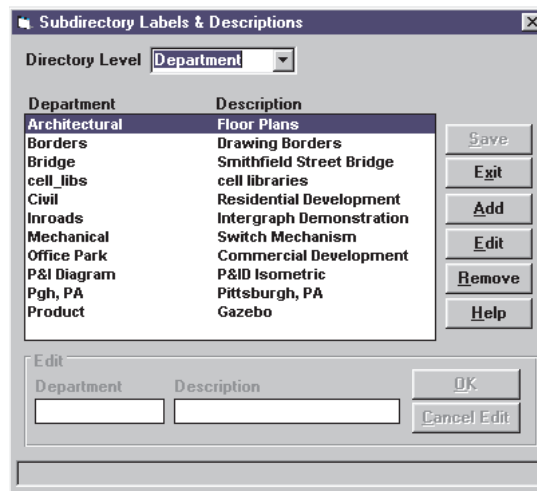
File Type/Filters

The File Type/Filters dialog box is the form in which the contents of the file filter drop down menus is defined. The items defined from this dialog box are used to in menus that appear on MassAdd and both versions of the Document Manager.

To open the File Type/Filters dialog box:

From the Administrator, choose Configure > File Type/Filters.
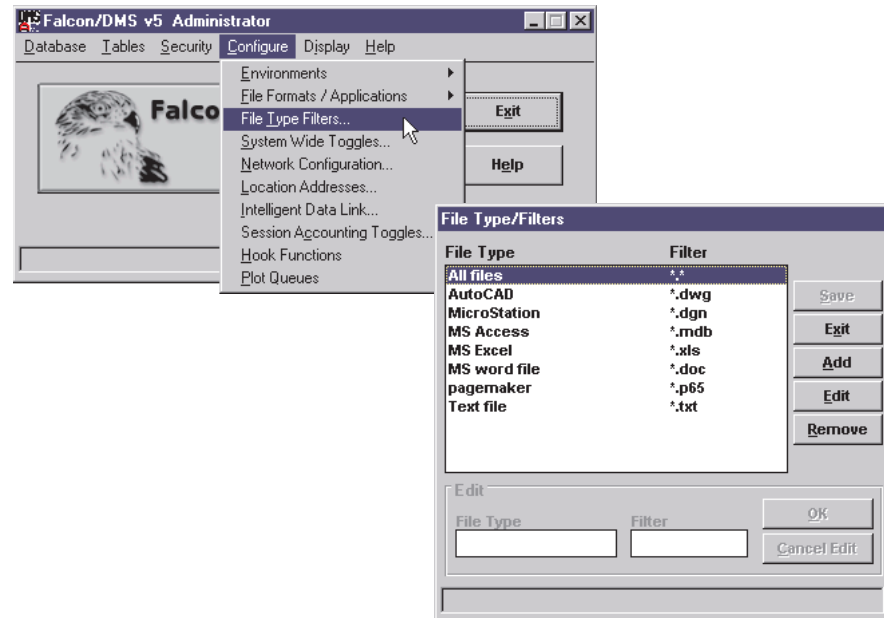
To add a File Type/Filter combination:

1. Press Add.

2. In the File Type field, enter an application name. Press <Tab>

3. In the Filter field, enter a file filter.

4. Press OK. Press Save and Exit.

To edit a File Type/Filter combination:

1. Double–click a File Type/Filter combination (or select and press Edit).

2. Edit as required.

3. Press OK. Press Save and Exit.

To remove a File Type/Filter combination:

1. Select a File Type/Filter combination.

2. Press Remove. Press Save and Exit.

Falcon/DMS

The Administrator

Chapter 6:
Security

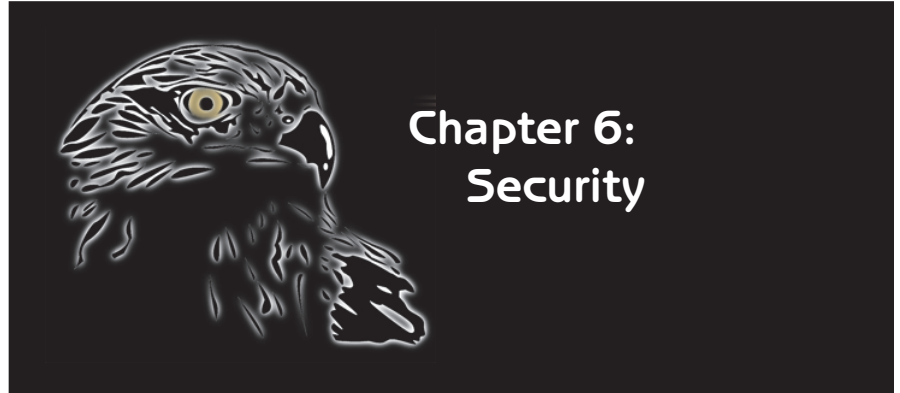Controlling each user's access to the documents on a network is a key reason why organizations invest in a document management system. However, the task of managing server-based documents can extend far beyond file open and check in control. Addressing the varied and complex issues of document access and operational privileges is an area in which Falcon/DMS excels, Let's first consider the issue of document access:

❏ Falcon/DMS can be configured to allow the data repository's operating system to control explicit directory and file permissions.

❏ By assigning a Document Access Status to a file or set of files, Falcon/DMS can use the database to authenticate file-level permissions. This allows you to increase the protection on documents without altering the explicit permissions assigned through the data repository's operating system.

❏ Multiple Environments can be used to customize the Master 'Project' Table. In other words, provide different project lists for different groups, departments, users, etc. (see Chapter 7).

❏ Project Security, a hook function that will be moved into the core product, can assume full responsibility for controlling directory access, shifting the job of assigning and authenticating permissions from the operating system to the database (see Appendix D).

Operational privileges, or the commands and database fields to which a user has access, is the other security addressed here. The Falcon/DMS administrator can define privileges ranging from the basic file open/check in capabilities of a junior user, to the expanded charter of a department supervisor. For example, you can easily limit the number of files a user can have checked out, control who can delete documents, or define the level of access a project supervisor has to the Falcon/DMS Administrator interface, to name a few.

Because the issue of 'groups' is central to the implementation of not only Document Access Status, but also to Multiple Environments and Project Security, we begin our examination of the security tools with an explanation of Accounts & Access Control, the environment in which you define user accounts and groups.

Accounts/Access Control

WARNING: Falcon/DMS is delivered with an ADMIN account that has been granted full operational privileges—until you thoroughly understand the role of the ADMIN account, **DO NOT** remove it from the Falcon/DMS account database or replace it with an equivalent. Also, **DO NOT** enable Access Control unless the ADMIN account, or its equivalent is present. If this warning is not heeded and Access Control is enabled absent the ADMIN account, further access to the Falcon/DMS Administrator is denied. (Enabling the Access Control Global Security Settings is discussed at the end of this section.)

Additionally, when Access Control is enabled for client workstations running under Windows NT, Falcon/DMS looks to match the current network login account to a name in its own account database. This means when Access Control is enabled and the ADMIN account is the only account with operational privileges sufficient to open the Accounts/Access Control environment, you will need to log on to the administrator's workstation as ADMIN. This network account does not have to be a domain administrator.

If you administer Falcon/DMS from Windows 95 or 98, the USERNAME environment variable set in AUTOEXEC.BAT must equal ADMIN—for Windows 95 or 98 you will not need a network login account named ADMIN.

The above serves simply as a warning against enabling Access Control or deleting the ADMIN account before Falcon/DMS is properly configured. Typically, an administrator will add his or her name to the Falcon/DMS account database and create an Access Control User (ACU) definition with full operational privileges. Only then will the ADMIN account be deleted.

Accounts/Access Control: An Overview

The Accounts/Access Control environment is used to build and maintain the account database, define groups and assign users to those groups, and to define operational privileges for individual users and groups. The term 'operational privileges' describes the specific commands to which a user or group has access. Operational privileges apply to the Document Manager (Windows and MDL) and the Administrator.

Operational privileges are defined and assigned through a technology called Access Control. There are two categories of Access Control:

❑ Access Control By User (ACU)—defines privileges for an individual user account. When an ACU definition is present, its privileges override those granted through the user's group membership. Presently, a user can belong to only one group. Unless otherwise specified, a new user's account is automatically assigned to the COMMON group.

❑ Access Control By Group (ACG)—defines privileges for users based on their member-

ship in a group. As just indicated, each user account is automatically assigned to the COMMON group. Additionally, any person using Falcon/DMS for whom an account has not been specifically defined, is automatically assigned to the COMMON group.

Falcon/DMS is delivered with the following items predefined:

❏ Two user accounts—ADMIN and TYPICAL.
❏ An ACG file, ADMIN.ACG, with full operational privileges. The ADMIN account is a member of the ADMIN group.
❏ An ACG file, COMMON.ACG, with limited operational privileges. The TYPICAL account is a member of the COMMON group.

The Access Control definitions established for users and groups are read when a user launches either the Windows or MDL version of the Document Manager or the Falcon/DMS Administrator. Typically, when the Administrator interface is available to any user other than the actual Falcon/DMS administrator, it is for the purpose of providing a greatly reduced command set to someone whose responsibilities may include, for example, adding a project to the Master 'Project' Table, or managing the list of System or File Format keywords. Only one key person or small group of key people should have access to the particularly security-sensitive environments in the Falcon/DMS Administrator—these would naturally include Database Maintenance, the environment allowing write access to all fields in the main database table, the database account from Display Database, Database Import, Global Security Settings, Accounts/Access Control, System Wide Toggles, and Session Accounting Toggles.

Let's now examine the process Falcon/DMS follows when Access Control is enabled.

❏ When a user opens, for example, the MDL Document Manager, Falcon/DMS searches the user account database for an account matching that user's login account name (for Windows NT), or an account matching the USERNAME variable, set in AUTOEXEC.BAT (for Windows 95 or 98), or finally, for UNIX, a variable set during the UNIX login or defined in the script file that launches MicroStation. (Note: Neither the Windows Document Manager nor the Falcon/DMS Administrator are available for UNIX).

❏ When an account matching the user's login account name is found, Falcon/DMS looks for an ACU definition. If an ACU definition is found, Falcon/DMS opens the MDL Document Manager with the command set and database access defined by the ACU.

❏ If an ACU definition is not found for the account name, Falcon/DMS reads the ACG definition for the group to which the user belongs. Falcon/DMS then opens the MDL Document Manager with the command set and database access defined by the AGC. If an account name exists, at the very least that user will belong to the COMMON group. In light of this, consider the following: Unless a user is to be specifically assigned to a group other than the COMMON group, it is not necessary to even define an account for that user in the Falcon/DMS account database—keep in mind, any person using Falcon/DMS for whom an account has not been specifically defined, is automatically assigned to the COMMON group.

❏ If the user does not have an account in the Falcon/DMS account database, Falcon/DMS then opens the MDL Document Manager with the command set and database access defined for the COMMON Group.

## The Accounts/Access Control Environment—a typical workflow

Defining the Access Control parameters for users and groups involves adding users to the Falcon/DMS account database, modifying the parameters defined for the COMMON group to meet your specific requirements, creating additional groups and assigning users to one of those groups, and, finally, if a particular user's Access Control parameters do not conform to those defined for any of the groups, to create and define an ACU for that user—and not necessarily in this order.

To provide a coherent explanation of the several operations required to set up an Access Control scheme, let's first outline the steps involved in a typical Access Control workflow. This workflow is followed by a explanation of each step in the process, in the order in which they are listed below.

1. Using the same name as your network login account, add yourself to the account database—this account will eventually replace the ADMIN account delivered with Falcon/DMS.

2. Create an ACU for your account and grant it full operational privileges. Because you are the Falcon/DMS administrator, only you account will be granted full operational privileges—think of your account as the 'super user' account. Assign your account to the ADMIN group. (Though your 'super user' account has privileges that will surpass those of even the ADMIN group, there are several benefits relating to Project Security available only to member of this group. See Appendix D.)

3. Add an account for an assistant administrator and assign that account to the ADMIN group.

4. Define the ACG privileges for the ADMIN group—these are the operational privileges granted to users who are assigned the responsibilities of assistant administrators.

5. Add an account for a senior designer and create a DESIGN group. For the DESIGN group, define operational privileges suitable for senior designers. For example, members of this group will be able to send documents to consultants using the Document Manager's Location Checkout command, as well as being able to add projects to the Falcon/DMS Administrator's Master 'Project' Table.

6. Define the ACG privileges for the COMMON group—these are the operational privileges that your organization deems to be the minimum necessary for a typical user.

7. Having defined your login account as the 'super user' it is now safe to remove the ADMIN account. As part of this step you will also remove the TYPICAL account.

8. On the Global Security Settings dialog box, enable Access Control.

To open the Document Status Access dialog box

On the Administrator, choose Security > Accounts/Access Control. The Accounts/Access Control dialog box opens.
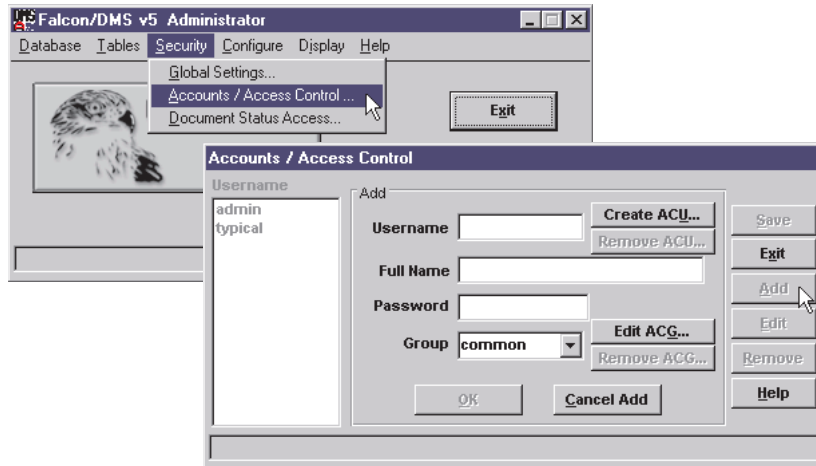
Step 1: Create the administrator's account

Using the same name as your network login account, add yourself to the account database—this account will eventually replace the ADMIN account delivered with Falcon/DMS.

To add the administrator's account:

1. On the Accounts/Access Control dialog box, press Add.

2. In the Add region, enter your network login account name. Press <Tab>.

3. [Optional] In the Full Name field, enter your full name. Press <Tab>.

4. [Optional] Enter a password. This option is not widely used as most administrators are reluctant to require their users to login more than once.

5. From the Group drop-down list box, select ADMIN. Though this account will be given an ACU definition that supersedes the operational privileges of the ADMIN group, there are several benefits relating to Project Security available only to members of the ADMIN group (see Appendix D).
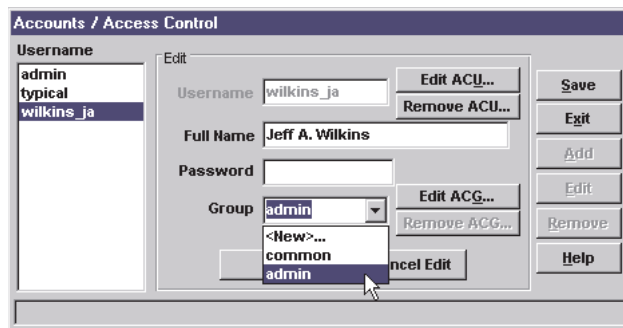
Because the next step relates to the current account, you can remain in the Add mode and proceed to create an ACU. Otherwise, you would press OK, Cancel Add, and then Save.

Step 2: Define an ACU for the administrator's account

Create an ACU for your account. Despite the existence of an ADMIN group and the administrator's default membership in COMMON group, your 'super user' account will have privileges that surpass those of the ADMIN group.

To define an ACU for the current account:

1. On the Accounts/Access Control dialog box, press Create ACU. The Create ACU dialog box opens.
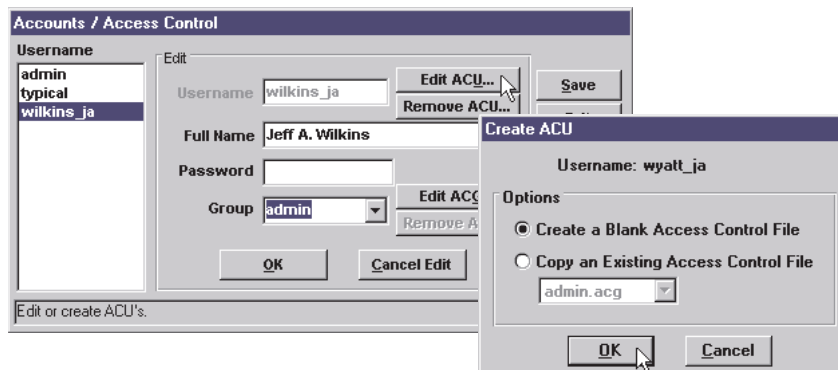
2. On the Create ACU dialog box, press OK to accept the default option—Create a Blank Access Control File.

   The second option—Copy an Existing Access Control File—allows you to use any existing ACG or ACU file as a template. Because users and groups are assigned privileges from the same pool of options, the files are identical except for their extensions. Note that an ADMIN.ACG exists. You could easily select this file and skip the remaining steps in this subsection. However, for the sake of explanation you will use a blank file.

3. On the Access Control Configuration dialog box, press Toggle All On/Off on the General Privileges tab. A breakdown of each tab and control on this dialog box appears in the next subsection.
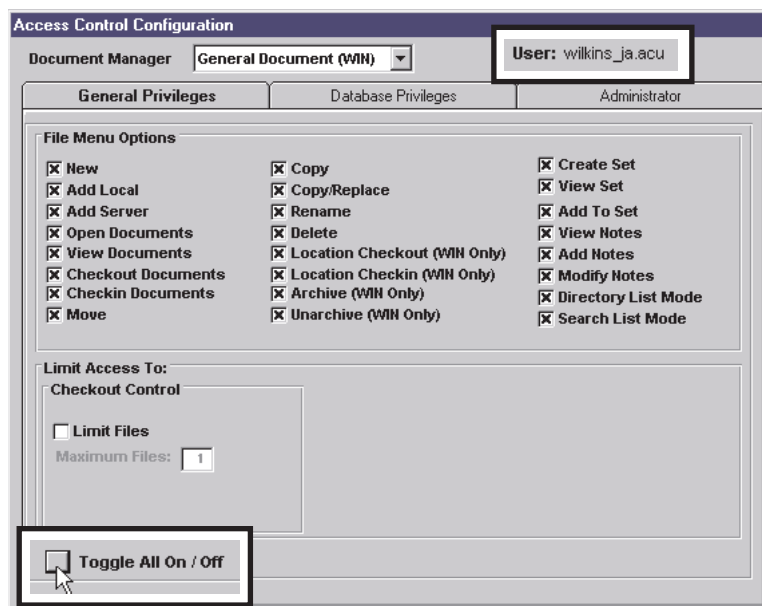
4. Select the Database privileges tab. Press Toggle All On/Off. From the Change 'Status' Level drop-down list box, select Full Edit.
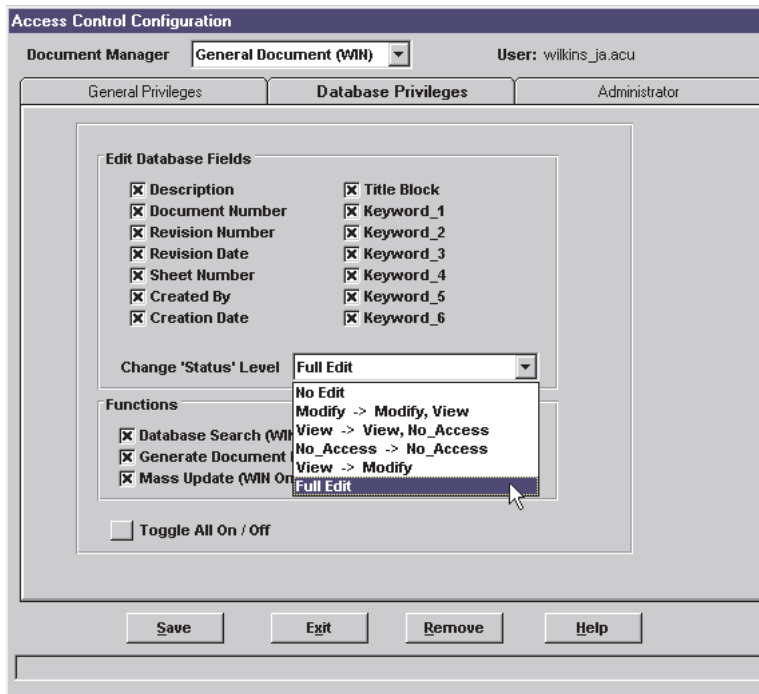
5. From the Document Manager drop-down list box, select MicoStation (MDL). Press Create ACU. The Create ACU dialog box opens. Repeat Steps 3 and 4.
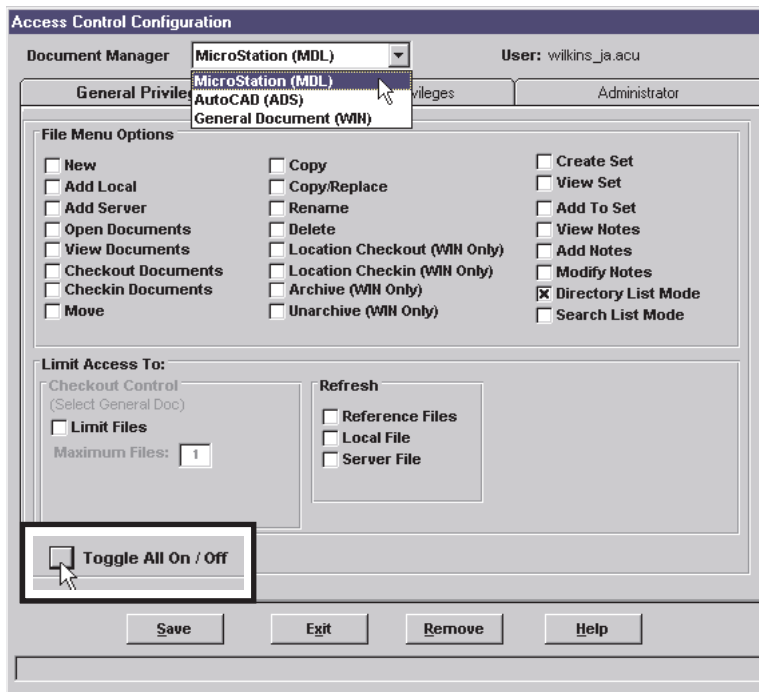
6. Select the Administrator tab. Press Toggle All On/Off.
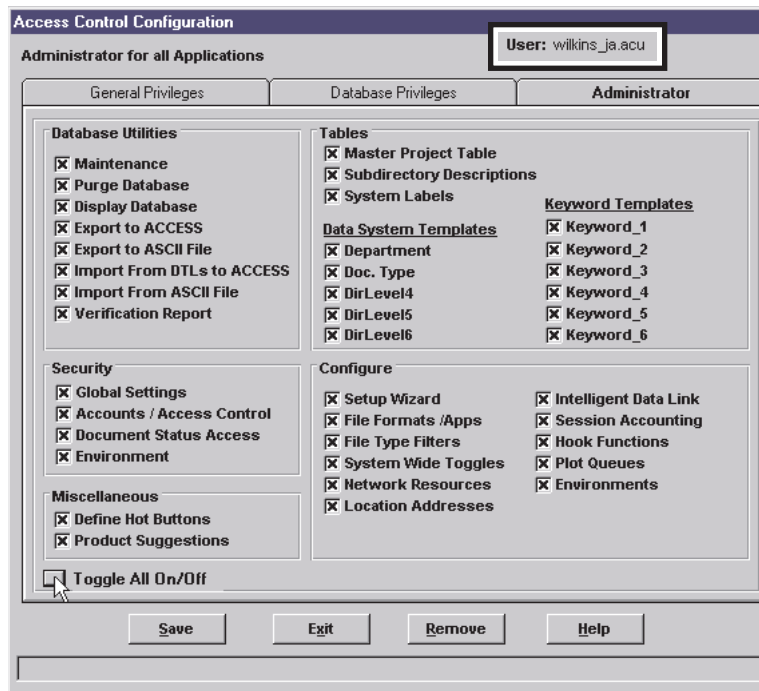
Press Save and Exit.

7. On the Accounts/Access Control dialog box, press OK, Cancel Edit, and Save. Because an ACU file now exists the Create ACU button is now labeled Edit ACU.

In the future, should you need to edit or remove this ACU, select your account, press Edit, then press Edit ACU or Remove ACU as needed.

8. Press OK, Cancel Add, and then Save.

The Access Control Configuration Dialog Box

The Access Control dialog box defines operational privileges found on all versions of the Document Manager as well as the Falcon/DMS Administrator. The operational privileges available to the Document Manager are defined for its three modules: the Windows Document Manager (WIN), MicroStation (MDL), and AutoCAD (ADS).

Document Manager:

The Document Manager drop-down list box displays the General and Database Privileges tab for the three versions of the Document Manager.

VERSION	DESCRIPTION
Windows	Displays the General & Database Privileges tabs for the Windows version (AutoCAD versions 13, 14, and 2000 interface with this version of the Document Manager.)
MicroStation	Displays the General & Database Privileges tabs for the MDL version.
AutoCAD	Displays the General & Database Privileges tabs for the ADS version (AutCAD version 12).

The General Privileges tab:

FILE MENU	OPTIONS
New	Permits a user to create a new document
Add Local	Permits a user to create database records for a document or documents located on the user's workstation and to place the record copy of the file/files on the data repository.
Add Server	Permits a user to create database records for a document or documents already located on the data repository. (Access to Directory List Mode is required.)
Open Documents	Permits a user to open and modify documents.
View Documents	Permits a user to view documents. Automatically enables or disables Copy Out (View).
Checkout Documents	Permits a user to checkout one or more documents.
Check In Documents	Permits a user to return checked out documents to the data repository.
Move	Permits a user to move a document.
Copy	Permits a user to copy a document.
Copy/Replace	Permits a user to replace an existing 'target' document with a copy of aa 'source' document by overwriting (replacing) the target document.
Rename	Permits a user to rename a document.
Delete	Permits a user to delete a document.
Location Checkout	[Windows Only] Permits a user checkout or copy out files to another location.
Location Check In	[Windows Only] Permits a user to check in files from another location.
Archive	[Windows Only] Permits a user to archive files.
Unarchive	[Windows Only] Permits a user to restore archived files.
Create Set	Permits a user to create a set.
View Set	Permits a user to view a set.
Add To Set	Permits a user to add to an existing set.
View Notes	Permits a user to view notes.
Add Notes	Permits a user to add notes to a document's database record.
Modify Notes	Permits a user to modify notes.
Directory List Mode	Permits a user to view the actual directories and files on the data repository. This view includes documents that have database records and documents that do not. Access to this mode is required for users permitted to add 'server-based' files to the database (see Add Server).
Search List Mode	Permits a user to view a database representation of the directories and files on the data repository.

The General Privileges tab (cont.):

LIMIT ACCESS TO:	DESCRIPTION
Limit Files	When enbled, activates the Maximum Files control. The MDL and ADS Limit Files control is enabled from this check box.
Maximum Files	Sets the 'not to exceed' number for the current group or user.

The Refresh Commands (MDL only):

REFRESH (MDL only)	DESCRIPTION
Reference Files	Permits users to update any or all of the reference files attached to the active drawing with a more current version of that file.
Local File	Permits users to automatically retrieve and open a 'fresh' version of the current file. Because the local file is overwritten by a fresh copy of the server-based original, any modifications made to the local file are lost.
Server File	Permits users to automatically refresh the server-based original with the current file without closing the current file. Please note that if the server file is unintentially refreshed, you will have to retrieve a new record version from backup. [Use this command with extreme caution.]

The Database Privileges tab:

EDIT DATABASE FIELDS	OPTIONS
Description	Permits a user to edit the Description field.
Document Number	Permits a user to edit the Document Number field.
Revision Number	Permits a user to edit the Revision Number field.
Revision Date	Permits a user to edit the Revision Date field.
Sheet Number	Permits a user to edit the Sheet Number field.
Created By	Permits a user to edit the Created By field.
Creation Date	Permits a user to edit the Creation Date field.
Title Block	Permits a user to edit the Title Block fields.
Keyword 1 - 6	Permits a user to edit the indicated (System) Keyword field/fields.

The Change Status Level control defines a user's ability to modify a document's Document Status Access condition. Document Status Access is explained in the final section of this chapter.

CHANGE STATUS LEVEL	OPTIONS
No Edit	Denies access to the Status field
Modify > Modify, View	Permits a user to change a Status with a 'modify' access code to a different Status that has either a 'modify' access code or a 'view only' access code. Once set to a new Status, the user cannot change the Status. Denies a user permission to change any Status to a 'no_access' Status.
View > View, No Access	Permits a user to change a Status with a 'view only' access code to a different Status that has either a 'view only' access code or a 'no access' access code. Once set to a new Status, the user cannot change the Status. Denies a user permission to change any Status to a 'modify' Status.
No Access > No Access	Permits a user to change a Status with a 'no access' access code to a different Status that has a 'no access' access code. Once set to a new Status, the user cannot change the Status. Denies a user permission to change any Status to a 'modify' or 'view' Status.
View > Modify	Permits a user to change a Status with a 'view only' access code to a different Status that has a 'modify' access code. Once set to a new Status, the user cannot change the Status. Denies a user permission to change any Status to a 'no_access' Status.
Full Edit	Grants full access to the Status field.

The Database Privileges tab (cont.):

FUNCTIONS	OPTIONS
Database Search	[Windows & MDL] Permits a user to perform a database search.
Generate Document List	[Windows & MDL] Permits a user to generate a list of the current directory contents showing filename, document number, and description.
Mass Update	[Windows only] Permits a user to edit the database records for multiple files. [*Access to this command should be limited to key personnel.*]

Administrator:

The Administrator tab contains all the Administrator options controlled through Access Control. With few exceptions, none of the Administrator options will be made available to users. One exception would apply to a project administrator who has the authority to create projects and their subdirectories on the data repository. Such a person would require access to the Master 'Project' Table for the purpose of, at the very least, adding the project name and defining the server/resource combination.

Database Utilities:

DATABASE UTILITIES	OPTIONS
Maintenance	Permits a user access to the Database Maintenance dialog box.
Purge Database	[Btree only] Permits the user to compress the Btree database.
Display Database	Permits a user to view the Primary Index Key and change the Database Account.
Export to ACCESS	[Btree only] Permits a user to export the Btree database to Microsoft Access.
Export to ASCII	Permits a user to export selected database fields to an ASCII text file.
Import from DTLs to Access	[Btree only] Permits the user import the daily transaction log (falcon.dtl) to an Access report database.
Import from ASCII	Permits a user to import a delimited ASCII file for the purpose of updating and/or creating database records.
Verification Report	Permits a user to run a database verification report.

Tables:

TABLES	OPTIONS
Master 'Project' Table	Permits a user access to the Master 'Project' Table dialog box.
Subdirectory Descriptions	Permits the user to define subdirectory descriptions.
System Labels	Permits a user to define data system labels.
Data System Templates	[DirLevel2 - 6] Permits a user to define and modify data system templates (subdirectories).
Keyword Templates	[System Templates 1-6] Permits a user to define and modify keywork templates.

Security:

SECURITY	OPTIONS
Global Settings	Permits a user access to the Global Settings dialog box.
Accounts/Access Control	Permits a user access to the Accounts/Access Control dialog box.
Document Status Access	Permits a user access to the Document Status dialog box.
Environment	Permits a user access to the Environments Security dialog box.

Configure:

| CONFIGURE	OPTIONS
Setup Wizard	Permits a user access to the Setup Wizard.
File Formats/Apps	Permits a user access to the File Formats/Applications dialog box.
File Type Filters	Permits a user access to the File Type Filters dialog box.
System Wide Toggles	Permits a user access to the System Wide Toggles dialog box.
Network Resources	Permits a user access to the Network Resources dialog box.
Location Addresses	Permits a user access to the Location Addresses dialog box.
Intelligent Data Link	Permits a user access to the Intelligent Data Link dialog box.
Session Accounting	Permits a user access to the Session Accounting Toggles dialog box.
Hook Functions	Permits a user access to the Hook Functions dialog box.
Plot Queues	Permits a user access to the Plot Queues dialog box.
Environment Configuration	Permits a user access to the Environment Configuration dialog box.

Miscellaneous:

MISCELLANEOUS	OPTIONS
Define Hot Buttons	Permits a user customize the Administrator by displaying up to 5 hot buttons.
Product Suggestions	Permits a user access to the Product Suggestions dialog box.

Step 3: Create an account for an assistant administrator

The account that identifies the assistant Falcon/DMS administrator will be created and assigned to the ADMIN group.

To add an account and assign the account to an existing group:

1. On the Accounts/Access Control dialog box, press Add.

2. In the Add region, enter the network login account name for the assistant Falcon/DMS administrator. Press <Tab>.

3. [Optional] In the Full Name field, enter the full name. Press <Tab>.
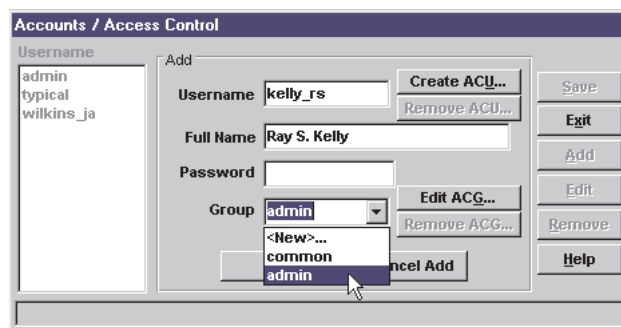
4. From the Group drop-down list box, select ADMIN.

   Because the next step relates to the current account, you can remain in the Add mode and proceed to edit the ADMIN Access Control Group definition. Otherwise you would press OK, Cancel Add, and then Save.

Step 4: Modify the operational privileges for the ADMIN group

Because the 'super user' defined in Steps 1 and 2 has full operational privileges and assistant administrators are to be assigned to the ADMIN group, it is necessary to modify the privileges defined for the ADMIN group.
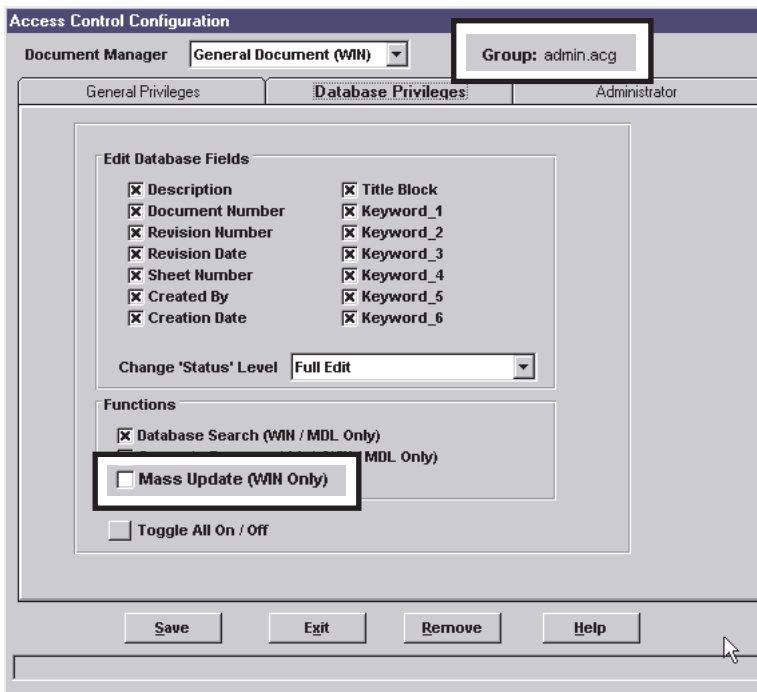
To modify the ADMIN group's Access Control Group definition:

1. Press Edit ACG. The Access Control Configuration dialog box opens.
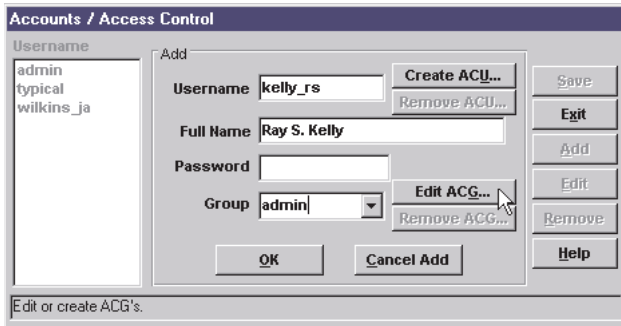

2. Modify the ADMIN.ACG definition as necessary. For example, consider limiting access to the following items to the 'super user' account:

   On the Database Privileges tab, deselect Functions > Mass Update (WIN only).

On the Administrator tab, deselect the following:

❏ Database Utilities > Maintenance
❏ Database Utilities > Display Database
❏ Database Utilities > Import From ASCII File
❏ Security > Global Settings
❏ Security > Accounts/Access Control
❏ Configure > System Wide Toggles
❏ Configure > Session Accounting
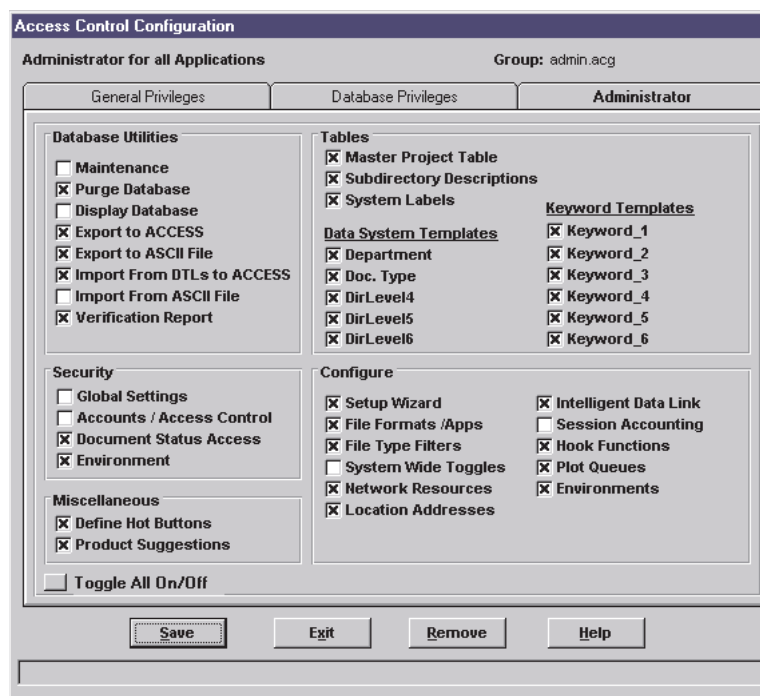
3. Press Save and Exit.

4. On the Accounts/Access Control dialog box, press OK, Cancel Add, and then Save.

Review

Completing 'Step 4—Modify the operation privileges for the ADMIN group' puts us at the halfway point of this example of a typical Accounts/Access Control workflow. Let's review what has been accomplished so far:

Step 1—you added an account to the Falcon/DMS account database using the same name as your network login account. This account will be configured as the main administrator account—the Falcon/DMS 'super-user'.

Step 2—you established an Access Control User (ACU) definition granting full operational privileges to the 'super-user' account.

Step 3—you added an account for an assistant administrator and assigned the account to the ADMIN group.

Step 4—you modified the operational privileges of the ADMIN group such that only the 'super user' account has access to the most security sensitive operations in Falcon/DMS.

## Step 5: Add a 'senior designer' account and a DESIGN group

After adding the 'senior designer' account and creating a DESIGN group for senior designers, group leaders, etc., you will define operational privileges suitable for this group. For example, members will be able to send documents to consultants using the Document Manager's Location Checkout command, and add projects to the Master 'Project' Table on the Administrator.

To add a 'senior designer':

1. On the Accounts/Access Control dialog box, press Add.

2. In the Add region, enter a network login account name defined for a senior designer. Press <Tab>.

3. [Optional] In the Full Name field, enter the designer's full name. Press <Tab>.

4. From the Group drop-down list box, select <NEW>. The Create ACG dialog box opens.
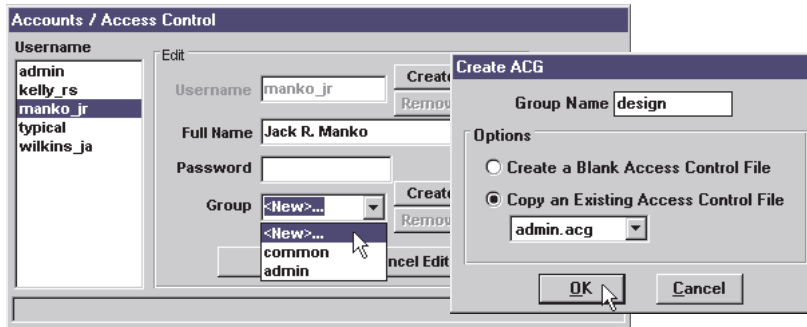
5. On the Create ACG dialog box, enter 'DESIGN' in the Group Name input field. Enable 'Copy an Existing Access Control File' and select ADMIN.ACG from the drop-down list box. The Configure Access Control dialog box opens.

6. On the Configure Access Control dialog box, select the Administrator tab. Press Toggle All On/Off. In the Tables region, enable the Master Project Table option.
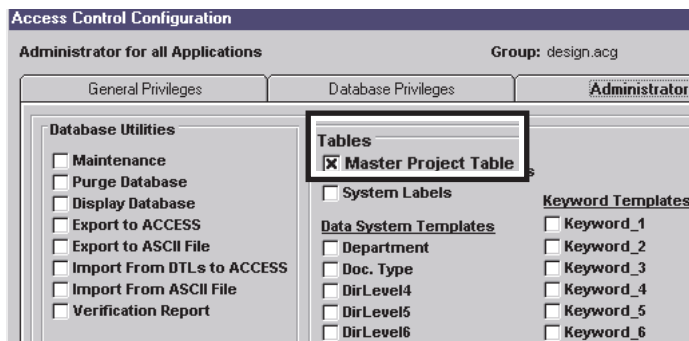
7. Press Save and Exit.

8. On the Accounts/Access Control dialog box, press OK, Cancel Add, and then Save.
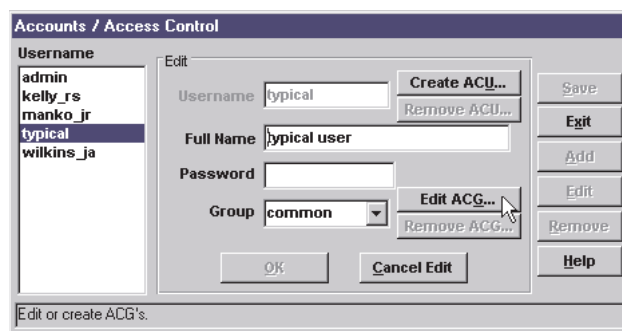
The COMMON Group

In most organizations users are commonly assigned to departments or groups. Falcon/ DMS uses this organizational construct to efficiently assign different sets of operational privileges to large numbers of people—this is accomplished with multiple ACG definitions. As you have seen, certain individuals sometimes require greater privileges than those assigned to their group—this is accomplished through ACU definitions. In most, if not all organizations, there will be a 'common' group that accounts for the greatest number of users, with much smaller numbers of users placed in 'elite' groups. Falcon/DMS accommodates this largest group by automatically placing each user who is not a member of an 'elite' group into its COMMON group. Planning how this COMMON group is implemented can be especially useful in terms of maintaining the account database. Remember, when a Falcon/DMS user is not specifically listed in the account database, that user is automatically assigned to the COMMON group. Given this, it is not necessary to define an account for every user. Rather, you need define an account for only those users belonging to an 'elite' group.

Step 6: Define operational privileges for the COMMON group

Define the ACG privileges for the COMMON group—these are the operational privileges that your organization deems to be the minimum necessary for a typical user.

To view and edit the COMMON Group privileges:

1. In the Username list window, double-click typical. Note that this user is assigned to the COMMON group.

2. Press Edit ACG. The Access Control Configuration dialog box opens.

Review the default settings for the COMMON group and modify as needed. If changes are made, press Save and Exit upon completion.

3. On the Accounts/Access Control dialog box, press Cancel Edit, Save and Exit.

Step 7: Remove the ADMIN and TYPICAL accounts

Having defined your login account as the 'super user'—by creating an ACU that has access to all operations and features—it is now safe to remove the ADMIN account. As part of this step you will also remove the TYPICAL account.
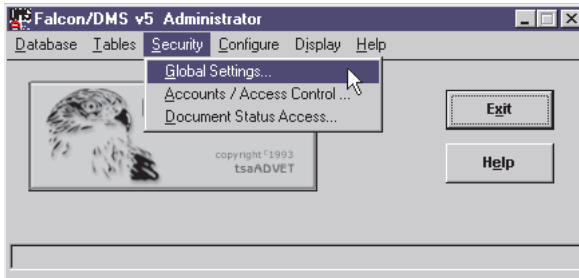
To remove an account:

1. In the Username list window, select ADMIN. Press Remove.

2. On the Confirm Remove message box, press OK.

3. Repeat for the TYPICAL account.

4. Press Save and Exit.

Step 8: Enable Access Control

To enable Access Control.

1. From the Administrator, select Security > Global Settings. The Global Settings dialog box opens.

2. Under Access Control Files, select Enable Access Control for Users/Groups (ACG/ACU).
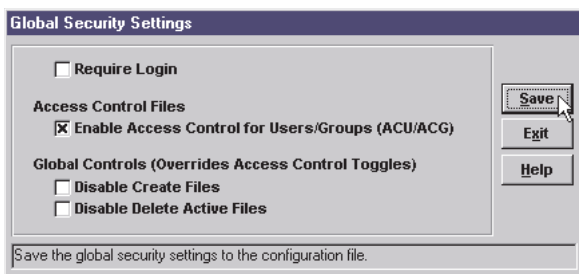
3. Press Save and Exit.

Additional Global Security Settings options

❑ Require Login: When enabled, users are required to login to Falcon/DMS in addition to logging on to the network. This option is not widely used as most administrators are reluctant to require their users to login more than once.

❑ Global Controls (Overrides Access Control Toggles): These controls are useful when the Falcon/DMS administrator needs to instantly halt the creation and/or deletion of documents

Document Access Status

By assigning a Document Access Status to a file or set of files, Falcon/DMS can use the database to set certain file-level permissions, allowing you to increase the protection on documents without altering the explicit permissions assigned through the data repository's operating system.

For example, a set of document comes under review. It is determined that during the review process, any access to the documents, either modify or view, be restricted to members of the REVIEW group. Any person not in the REVIEW group needs to be told that the documents cannot be accessed and for what reason. From the operating system, the explicit file permissions for this set of documents have been set so only members of the DESIGN group can modify the documents, while all other users are permitted to view the documents. This now has to change, immediately, and the network administrator is not available to set the new file permissions. The answer—Document Status Access.

A Document Status Access has three components: a status name, a group assigned to the status, and an Access Code. There are three Access Codes: modify, view_only, and no_access.

There are just three steps to implement a Document Status Access change for the example above:

1. Create a document status. Let's call it 'under_review'. Then assign the REVIEW group and the modify access code to the 'under_review' status.

2. Create a second document status. Give it the same name as the first—'under_review'. Then assign the <default> group and the no_access access code to this second 'under_review' status. Finally, define a message informing all users not in the REVIEW group why they cannot access any document with an 'under_review' status.

3. From the Document Manager, select the document whose status you want to set to 'under_review', select 'under_review' from the Status drop-down list box, and press Save. For multiple files, use the Windows Document Manager's Mass Update command to assign the new status.

While in principal they are similar, do not confuse the <default> group in Document Access Status with the COMMON group in Access Control. The <default> group is designed to encompass all groups that are not specifically assigned to a Document Access Status. This means that when you have two or more like-named statuses with different access codes—for instance, the 'under_review' example above—you are not required to define a like-named status for each and every group.

What the user sees:

When a member of the REVIEW group opens, views, checks out, or copies out one or more 'under_review' documents, Falcon/DMS delivers the documents as requested.

When a user not in the REVIEW group attempts any operation on an 'under_review' document, Falcon/DMS displays the following message:
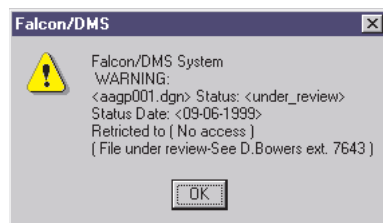
An Overview

Falcon/DMS can enhance the data repository's security constructs by assigning a Document Access Status to individual files or sets of files. While working in concert with the security accounts manager, Document Access Status uses the database to provide an extra layer of file-level permissions that is very easy to manage. By involving the database in file-level security, you can increase the protection on documents without altering the explicit file permissions assigned through the data repository's operating system. As long as users access the data repository through Falcon/DMS, Document Access Status can, in effect, change a modify document to view_only or no_access, or a view_only document to no_access. It is important to note that Document Access Status cannot, however, *increase* any file's access level. In other words, Falcon/DMS can never make a no_access document, as defined by the data repository's operating system, a modify or view_only document, or change a view_only document to modify. Consider the following example:

When responsibility for a set of documents is passed from one group to another for the next stage of the design process, let's say from Group A to Group B, access to these documents by Group A, the group releasing the documents, should now be restricted to view_only. As long as the only way to access these documents is through Falcon/DMS, you can use Document Access Status to set the access mode for Group A to view_only without altering the explicit file permissions set through the operating system . Thus, when a member of Group A, with operating system privileges sufficient to modify documents, attempts to open a file that has advanced to the next step in a 'workflow', that designer is not able to alter the document without receiving authorization through the database. Obviously, restricting access to the data repository through Falcon/DMS, and only through Falcon/DMS, is key.

An Access Status can also be defined with different access levels for different groups. For example a 'Review' status could be configured to restrict members of the common, or default, group to no_access, while members of the management group as well as designers within the reviewing group would have modify privileges.

Additionally, Falcon/DMS can control who is able to change a Document Access Status and how. Please refer to Accounts/Access Control, Database Privileges tab, Change Status Level.

To open the Document Status Access dialog box:

On the Administrator, choose Security > Document Status Access.
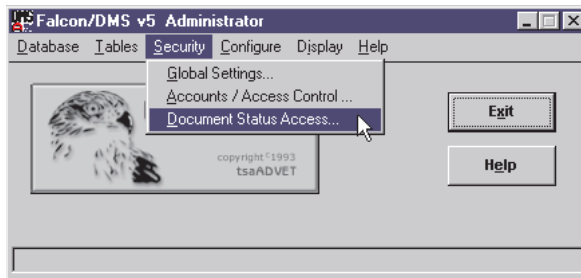
The Basics

The Document Status Access dialog box is used to define Statuses (e.g. active, approval, release, etc.), assign an access code (i.e. modify, view_only, no_access), and define messages that explain a document's access state.

To add a Document Status:

1. Press Add.

2. In the Status field, enter up to 15 characters.

3. From the Group drop-down list box, select a group.

   When a status is to apply to all users, select <default>.

   The Group option reads the Group table defined in the Accounts/Access Control environment.

4. From the Access Code drop-down list box, select one of the three options: modify, view_only, or no_access.

5. If you want users to have access to a document's database record, check the DB Record Access check box.

6. [Recommended for view_only and no_access statuses] In the Message field, enter up to 40 characters.

7. Press OK. Repeat Steps 2 through 6 as needed.

8. Press Cancel Add, Save, and Exit.

To edit a Document Status:

1. Double-click on an entry in the list box.

2. Modify the Group, Access Code, and Message as needed.

3. Press OK and Save.

To remove a Document Status:

1. Select an item in the list window.

2. Press Remove.

3. Press Yes on the alert palette.

4. Press Save.

Assigning multiple groups to a single status

As mentioned previously, a specific Access Status can be assigned to multiple groups, with different access levels defined for each group.

For example, the status 'Internal Review' is assigned to three groups, Group A, Group B, and <default>. The access code for Group A is view_only and no access is permitted to the database records for 'Internal Review' documents. The access status for Group B, on the other hand, is modify for files with this status and full access to their database records. All remaining users, that is users who are members of neither Group A nor Group B, will not be able to access any document with an 'Internal Review' status. For the purpose of Document Status Access these remaining users are members of the <default> group.

Document Status Access

Status	Group	Access Code	DB Access	Message
active	<default>	modify	yes	
complete	<default>	view_only	yes	
construction	<default>	modify	yes	
indexed	<default>	modify	yes	
internal review	<default>	no_access	no	
internal review	group a	view_only	no	
internal review	group b	modify	yes	
off-line	<default>	no_access	yes	
repro	<default>	modify	yes	
verified	<default>	view_only	yes	

Save
Exit
Add
Edit
Remove
Help

Edit
Status
Group
Access Code
□ DB Record Access
OK
Message
Cancel Edit

To define the 'Internal Review' status for multiple groups:

GROUP A. - view_only

1. Press Add.

2. In the Status field, enter internal review.

3. From the Group drop-down list box, select a GROUP A.

4. From the Access Code drop-down list box, select view_only.

5. Do not check the DB Record Access check box.

6. In the Message field, enter up to 40 characters.

7. Press OK.

GROUP B. - view_only

1. Press Add.

2. In the Status field, enter internal review.

3. From the Group drop-down list box, select a GROUP B.

4. From the Access Code drop-down list box, select modify.

5. Check the DB Record Access check box.

6. Press OK.

<default> - no_access

1. Press Add.

2. In the Status field, enter internal review.

3. From the Group drop-down list box, select <default>.

4. From the Access Code drop-down list box, select no_access.

5. Do not check the DB Record Access check box.

Falcon/DMS

The Administrator
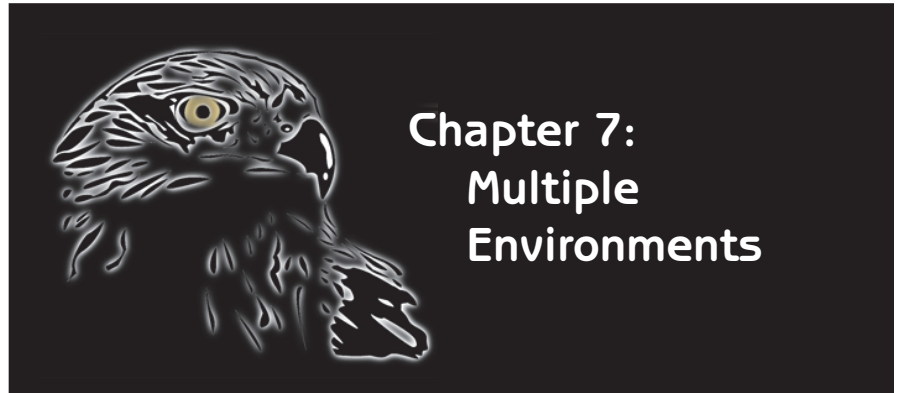
Chapter 7:
Multiple
Environments

The Multiple Environments feature allows different groups, departments, or individuals to work with a set of control parameters and a user interface configuration specific to the unique requirements of each.

Simply put, Multiple Environments offers different ways of organizing the configurations files that effect the many controls and option lists that appear on the Windows and MDL Document Managers.

An environment definition can use existing configuration files and tables (global definitions) for certain controls and build unique configuration files and tables for others. Also, Falcon/DMS can use any of the existing global definitions as a template for a new definition, making the Multiple Environments feature easy to configure.
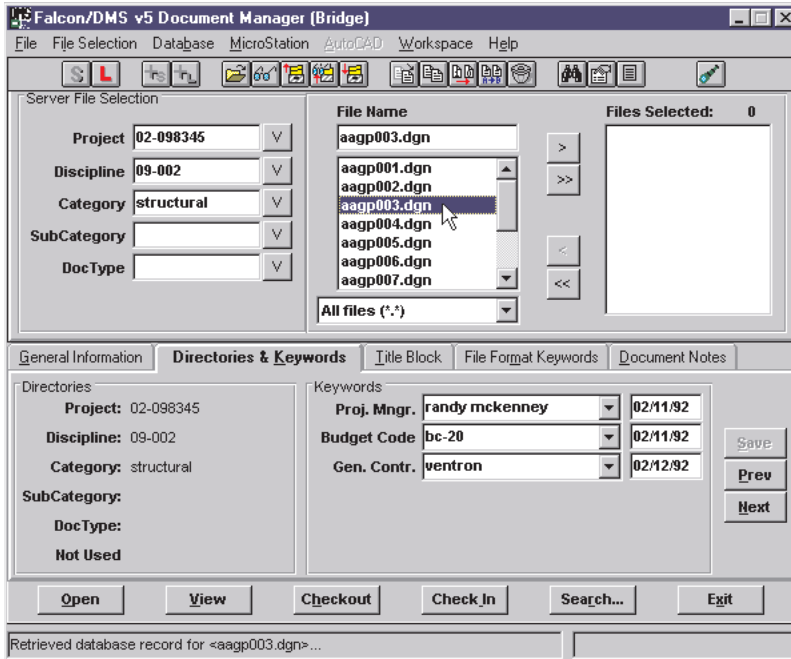
Multiple Environments

Examples of how Multiple Environments effect the appearance of the user's interface appear below.

The Document Manager for the [Design] environment uses EA#, Highway#, and SubCategory for its directory labels and has its own 'EA#' table (Master 'Project' Table). It has six system level Keyword Categories with unique tables. Compare this with the Document Manager for the Bridge environment.
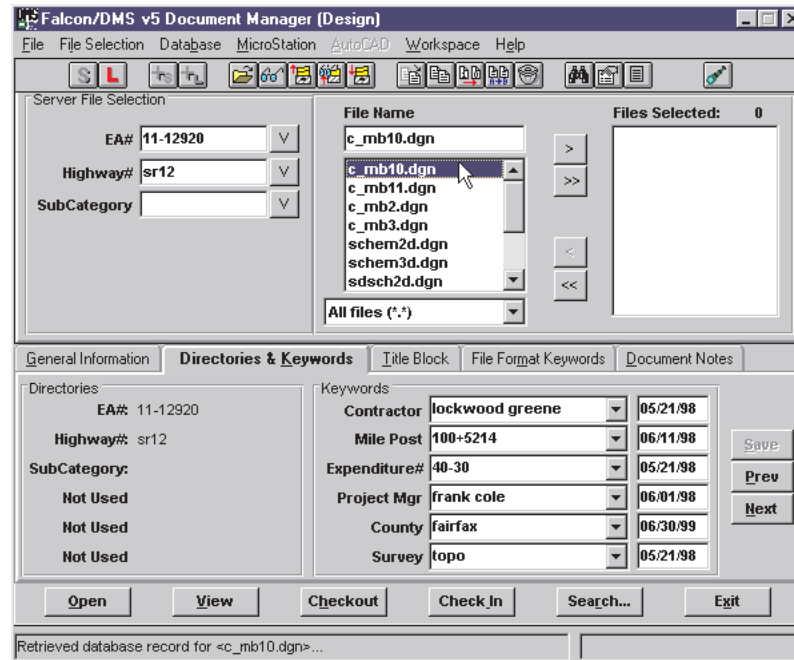
The Document Manager for the [Bridge] environment (below) uses Project, Discipline, Category, SubCategory, and DocType for its directory labels and has its own Master 'Project' Table. It has three system level Keyword Categories with unique tables.

Multiple Environments: Behind the Scenes
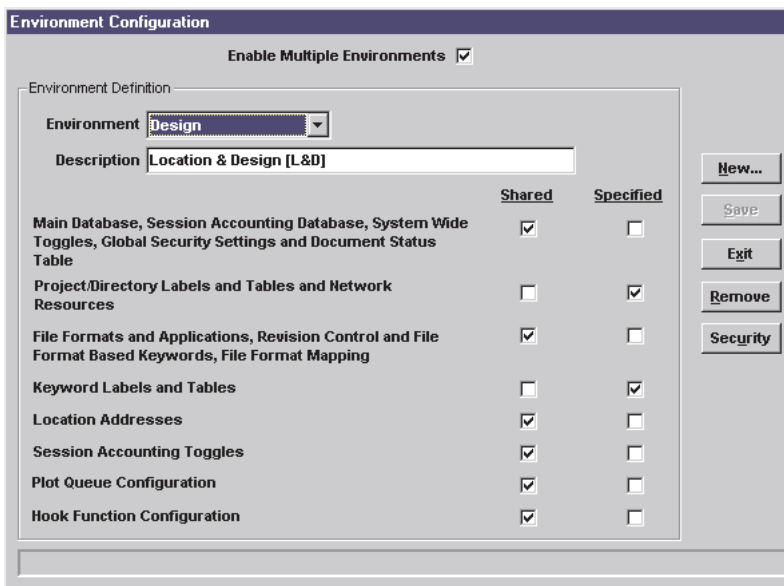
When you create a new environment you can share the default configuration files or specify a complete set of new configuration files.

When you define a new environment, Falcon/DMS creates a subdirectory with the name of that environment under <falcon_sdir>/cfg. Any configuration files unique to the environment are placed in this subdirectory. For example, the Environment Configuration dialog box for the Design environment (above) shows Design has five 'shared' and two 'specified' configuration groups. 'Specified' configuration files are unique to that environment. For 'specified' configuration groups you can copy the 'shared' files or create blank versions of the necessary files. The configuration files for any specified group are placed in <falcon_sdir>/cfg/environment_name.

After you define the environment you edit the 'specified' configuration files by setting the Administrator's environment to the new environment. Then you edit the files pertaining to the 'specified' configuration groups.

GROUP	COMMAND	LOCATION
Group 1	Main Database & Session Accounting Database	Database > Display Database > Set Account
	System Wide Toggles	Configure > System Wide Toggles...
	Global Security Settings	Security > Global Settings...
	Document Status Table	Security > Document Status Access...
Group 2	Project Directory Labels & Tables	Tables > System Labels/Templates > Directory Labels /Templates...
	Network Resources	Configure > Network Configuration...
Group 3	File Formats & Applications	Configure > File Formats/Applications > Edit Global Configuration...
	Revison Control	Configure > File Formats/Applications > Edit Global Configuration... --- Select & Edit a File Format --- Press Revision Control
	File Format Based Keywords	Configure > File Formats/Applications > Edit Global Configuration... --- Select & Edit a File Format --- Press File Format Keywords
Group 4	System Keywords & Tables	Tables > System Labels/Templates > Database Keyword Labels/Templates...
Group 5	Location Addresses	Configure > Location Addresses...
Group 6	Session Accounting Toggles	Configure > Session Accounting...
Group 7	Plot Queue Configuration	Configure > Plot Queues > Application...
Group 8	Hook Function Configuration	Configure > Hook Functions > Edit Server...

Enabling & Defining Multiple Environments

The following steps demonstrate the procedure used to create an environment. The example shows the creation of the Design environment pictured on the previous page. This environment has two 'specified' groups.

Step 1: Enable Multiple Environments

1. From the Administrator's menu bar, select Configure > Environments > Configuration.
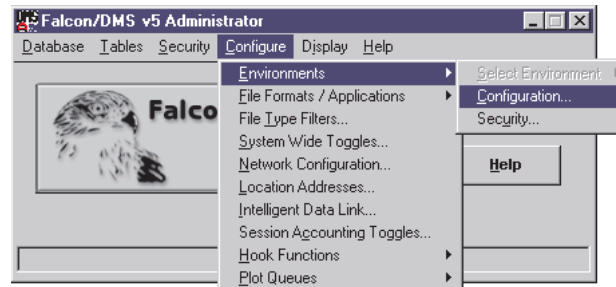

2. On the Environment Configuration dialog box, click the Enable Multiple Environments check box.
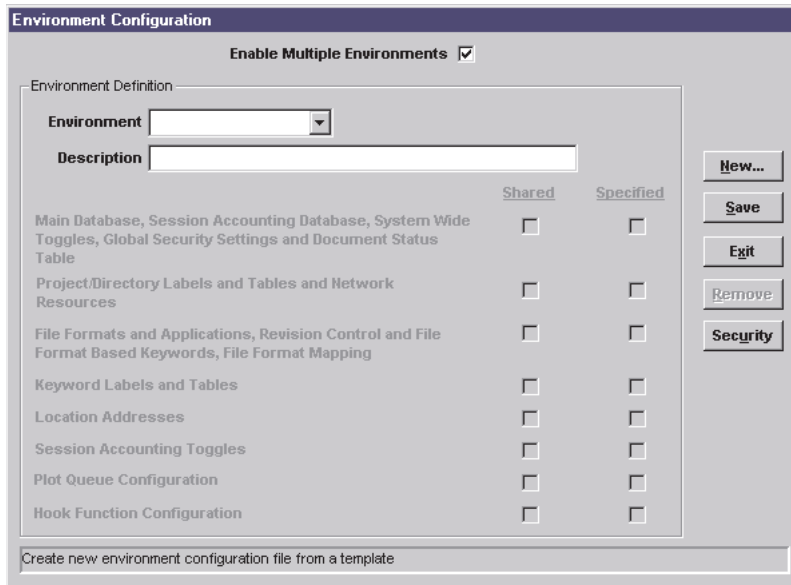
Step 2: Create the 'Master' Environment

The 'Master' environment is one in which all configuration categories are set to 'shared', thereby making it possible to edit the default set of configuration files without disabling the Multiple Environments feature.

1. Press New. The Create New Environment Configuration dialog box opens. Enter a name and a description for the master environment. Press OK.
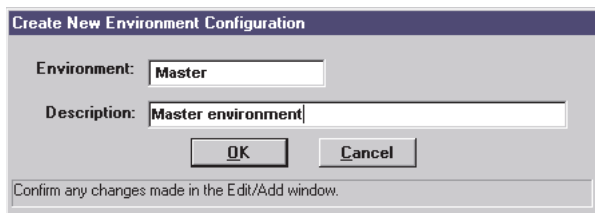
2. By default, the Shared option for all configuration categories is enabled; they will remain as shared for the Master environment. Press Save.

Step 3: Create a 'Production' environment called Design

A 'production' environment is considered to be any environment that is assigned to specified users or groups. A 'production' environment is usually comprised of a mix of shared and specified configuration groups, or entirely of specified configuration groups.

1. Press New. The Create New Environment Configuration dialog box opens. Enter a name and a description for a production environment. This example uses the Building/Design (BldgDsgn) environment. Press OK.

2. Because this is a typical production environment, certain configuration groups are 'specified'. The Design example has two 'specified' configuration groups. The first is the 'Project/Directory Labels and Tables and Network Resources' group. When the Specified check box is enabled for this group a second Environment Configuration dialog box opens.

❏ To create blank configuration files for the selected category, press the 'Create "blank" configuration file(s)' radio button.

❏ To copy and edit copies of the default configuration files for the selected category, press the 'Use shared configuration file(s) as template(s)' radio button.

Press OK.

3. Specify other configuration categories and choose a blank or shared source. Press Save.

When you save a new environment you are prompted to define its security parameters, that is what groups or users have access to the environment. Please refer to Environment Security (following Step 4) for detailed instructions on how to define security access for an environment.

Step 4: Configure the 'Production' environment

After defining the structure of an environment you must edit the contents of the 'specified' configuration files. In the Design example the 'Project/Directory' and 'Keyword Labels and Tables' configuration groups are 'specified' and can be customized (edited) from the Falcon/DMS Administrator. Use the following procedure as a general guide:

1. Select Configure > Environments > Select Environment > Design from the Administrator's menu bar.

    The Falcon/DMS Administrator's title bar now identifies the current environment as bldgdsgn.

2. From the Administrator, define the configuration options for those items associated with the 'specified' configuration groups.

Environment Security

Environment Security gives administrators the ability to control access to the different configurations defined within Multiple Environments. With Environment Security you can assign these specific configurations to different groups within the organization. Also, Environment Security can be used to assign specific configurations to individuals, regardless of his or her group membership. (This is particularly useful when Multiple Environments is used to define different Master 'Project' Table configurations and/or keyword schemes.)

Central to Environment Security is Accounts/Access Control. Though Access Control does not have to be enabled, to utilize Multiple Environments a User and/or a Group database must be defined. It is this database that you use to grant access to the different configurations defined in Multiple Environments.

Consider the following scheme

A transportation engineering organization, comprised of five engineering departments (Design, Bridge, Survey, Hydraulics, and Traffic) and an engineering support group, has eighty active projects. Only the engineering support group and the Survey department require access to all the projects. However, one person within each of the remaining four departments must also be granted full access to the table of active projects.

Requirements for the example environment scheme

Security > Accounts/Access Control:

1. Define six groups within Accounts/Access Control: Design, Bridge, Survey, Hydraulics (Hydro), Traffic, and Engineering Support (Support).
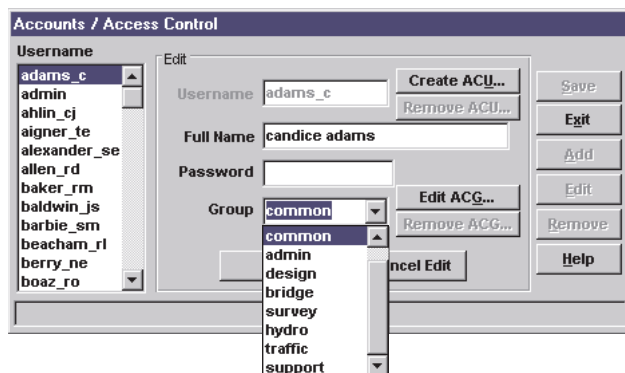
2. Define an individual account for each person who will be granted full access to the table of active projects. Then create an ACU file for each of these accounts.

Environments > Security... :

The Environment Security dialog box is the form used to grant users and groups access to one or more environments. The Environment Security dialog box can be opened from the Administrator's menu bar or from the Environment Configuration dialog box.

To grant access to a single environment:

1. From the Administrator, select Environments > Security... or from the Environment Configuration dialog box, press Security.
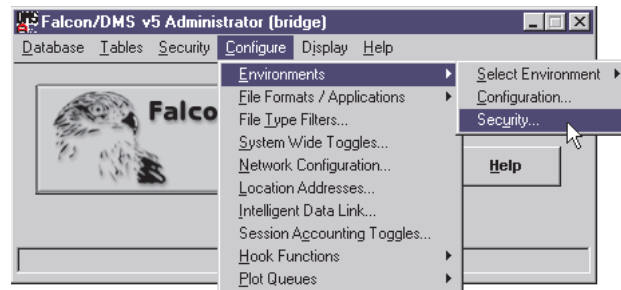
2. From the Environment Security dialog box, select a User/Group for which access to a single environment will be granted.

3. In the Access Denied list window, double-click on the environment to which the selected user/group will have access. The environment appears in the Access Allowed window.

4. From the Default Environment drop-down list box, select the environment.

5. Press Save.
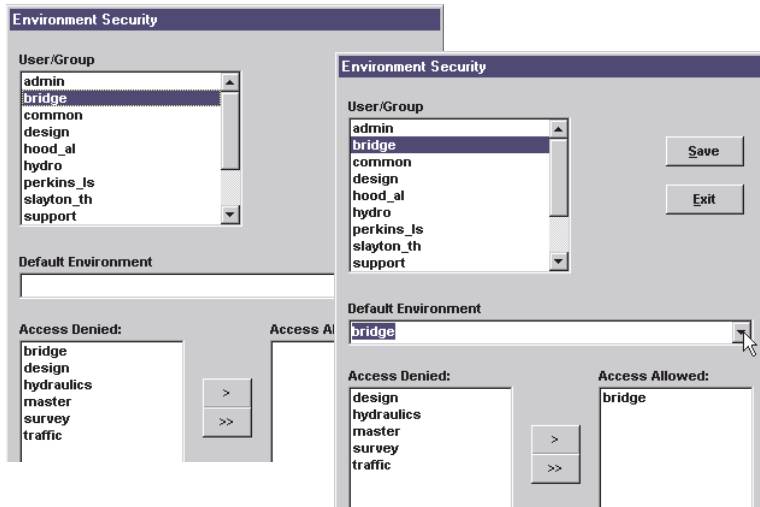
To grant access to all environments:

1. From the Administrator, select Environments > Security… or from the Environment Configuration dialog box, press Security.

2. From the Environment Security dialog box, select a User/Group for which access to all environments will be granted.

3. Press **>>** to select all the environments. The environments appear in the Access Allowed window.

4. From the Default Environment drop-down list box, select an environment. In the example the Master environment is chosen.

5. Press Save.

Multiple Databases and Environments

A powerful feature of Multiple Environments is the ability to access two or more databases in a manner that is completely transparent to the user.
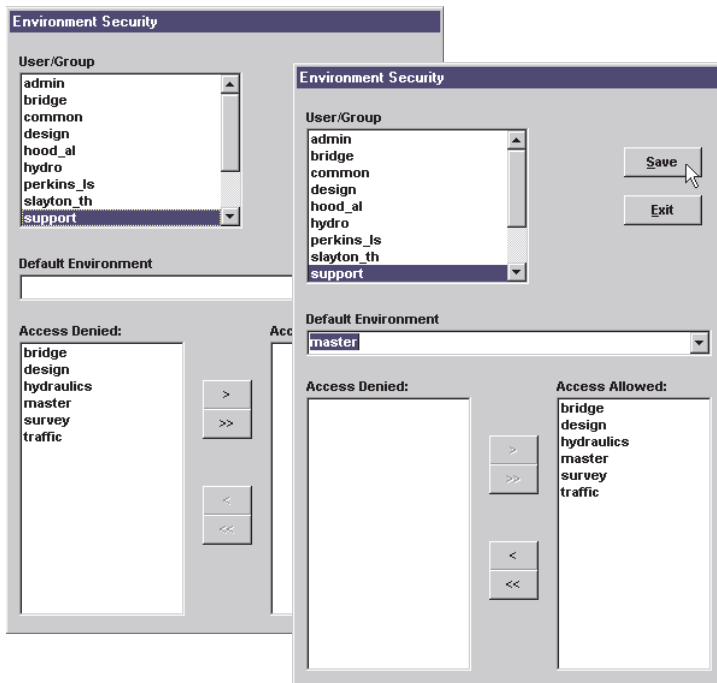
Consider the following scheme

Two environments, each referencing a different database.

A Master environment that uses shared configuration files for all the environment configuration groups. This environment accesses the default database which is indexed on unique filenames (primary index key = filename).

A second environment, created for a specific client, uses specified configurations for Groups 1 and 2, which include the Main Database and the Master 'Project' Table. The Master 'Project' Table contains only the projects for this client; these projects do not appear in the shared project table. The database for the client environment is indexed on <Directory Level 1, Directory Level 2, Filename>, allowing for three levels of filename uniqueness.

Requirements for the example environment scheme

Database:

1. Define two databases, each with a unique login account.

2. Define two ODBC data sources.

Falcon/DMS:

1. Define a Master environment that uses shared configurations for all the configuration groups.

2. Define a second environment, the client environment, that uses specified configurations for Groups 1 and 2.

3. When the 'Specified' option for Group 1 (Main Database, etc.) is selected, you must choose the 'Use shared configuration file(s) as template(s)' option. Press OK.

   When the 'Specified' option for Group 2 (Project Directory Labels & Tables, etc.) is selected, it is recommended that you choose the 'Create default/blank configuration file(s)' option. (Use MassAdd to populate the blank Master 'Project' Table for the client environment.)

4. From the Administrator, set the client environment as the current environment.

5. From the Administrator, select Database > Display Database...

6. From the Display Database dialog box, press Set Account...

7. From the Database Account dialog box, press Edit. Enter client Data Source Name and the account and password for the client database. Click Save and Exit.

8. Close and reopen the Administrator; select Database > Create Database. On the Create Database dialog box, set the primary index key to match your requirements and press Create Tables.

Falcon/DMS

The Administrator

Chapter 8:
Database
Maintenance
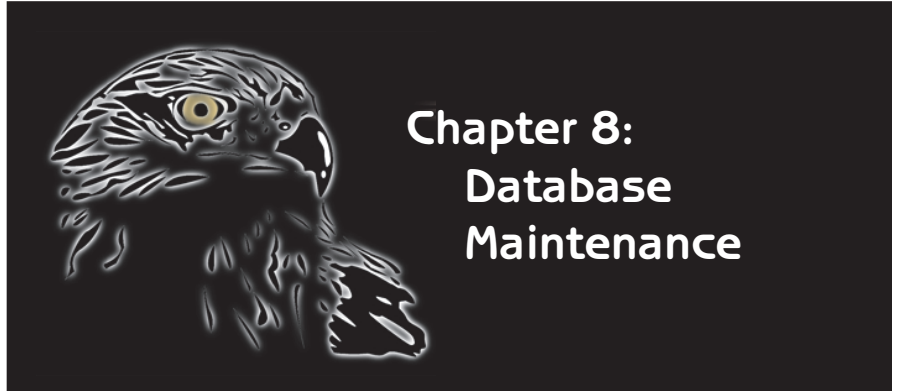
The focus of this chapter is on the database administration tools.

The Database Maintenance dialog box is an environment in which the Falcon/DMS administrator has access to every field in the main database table (FALCON50), and provides a way to easily address database discrepancies on a record-by-record basis. Should a document's database record not reflect the actual availability of that document, for example, re synchronizing the record with the document is accomplished by modifying the Availability Status setting in the General File Info environment.

The Database Export To ASCII command provides an efficient way to generate a delimited ASCII file containing information retrieved from selected database record fields. The Database Import From ASCII command provides a method to easily populate records with existing metadata.

A Database Verification enables an administrator to identify any discrepancies existing between the Falcon/DMS database and data repository. This operation is able to identify both records for which repository-based files are not found, and any repository-based files for which database records do not exist.

Using the Database Verification Report as part of the MassAdd phase of an implementation is a particularly effective way to identify files that have not been added to the database.

The Database Maintenance dialog box

To open the Database Maintenance dialog box:

From the Administrator, select Database > Maintenance.
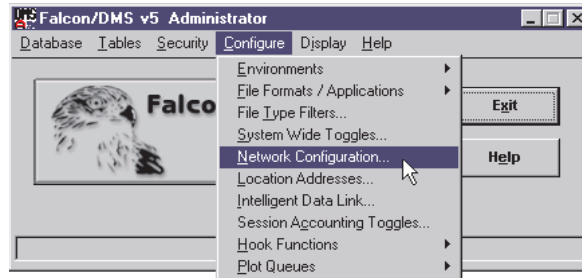
The Database Maintenance dialog box provides access to every field in each document's database record, as well as the ability to view a file's recent access history.

Retrieving database records:

The most efficient way to retrieve a database record is to use Browse to locate and select the electronic file on the data repository server. When a file is identified, Database Maintenance then retrieves and displays the file's database record. Please note that the Browse function is not available for the FTP network configuration—you can, however, specify UNC or Mapped Drives as a local network configuration without affecting how other Falcon/DMS clients see the network.

To use the Browse option:

1. On the Database Maintenance dialog box, press Browse. The Browse dialog box opens.
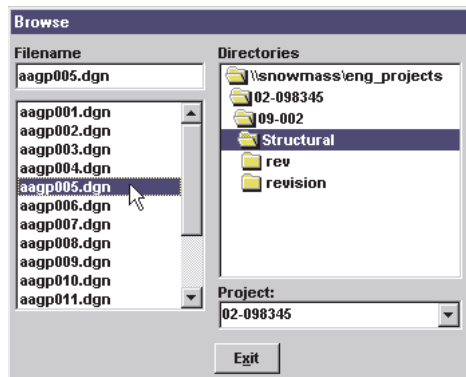
2. On the Browse dialog box, select the 'project' from the Project option menu.

3. In the Directories list window, double-click on the desired source directory. In the Filename list window, select a filename. The file's record displays in the Database Maintenance dialog box.

   The Browse dialog box remains open—position the Browse form so that is not obscured by the Database Maintenance dialog box and select additional files as needed.

4. To close the Browse form, press Exit.

There are several other ways to retrieve database records directly from the Database Maintenance dialog box that are dependent upon a filename's level of uniqueness. These options require that you, at the very least, key in a filename.

❏ When the Primary Index Key is set to Filename, you can key in a document's filename in the Filename field and press <Enter>.

❏ When the Primary Index Key is set to DirLevel1;Filename or greater, and a particular filename is used only once within the database, you can key in that document's filename in the Filename field and press <Enter>.

❏ When the Primary Index Key is set to DirLevel1;Filename or greater, and you wish to retrieve the database record for a filename used more than once, you can specify the location from the File ID text list boxes and then key in the filename.

The 'General File Info' Tab:

GROUP	FIELD	DESCRIPTION
Document Info	Sheet #	Drawing sheet number <4 character maximum>; user defined
	Rev #	Revision number; <5 characters>; user defined
	Revision Date	Revision date; automatically assigned
	Description	Document description <80 characters>; user defined
	Status	Drawing sheet number; <15 characters>; user defined
	Status Date	Status date; automatically assigned
Server Info	Availability Status	On Server: Document not checked out.
		On Workstation: Document checked out.
		At Location: Document sent to another location. Location Checkout
		Marked Deleted: Database record marked, actual document deleted
		Archived: Document archived through Archive operation.
		Archived Pending: Archive operation scheduled.
		Unarchive Pending: Unarchive operation sheduled.
		In Transit: Database not successfully upated
		Reserved by User: Document remains on server while checked out
		Associated: Document checked out as part of File Association operation
	Server	Identifies the document's data repository
	Resource	Identifies the resource on the document's data repository
	Directory	Identifies the document's data repository path
Workstation Info	User	Identifies the last or current user to whom the document was checked out.
	Station	Identifies the workstation for the user.
	/Date	Indicates the data on which the document was checked out.
	Local Filename	Indicates the local filename.
	Main Local Dir	Indicates the local working directory.
Location Info	Location	Identifies the location to which the document was checked out through the Location Checkout operation.
	Date	Identifies the date of the Location Checkout operation.
	Approval	Identifies the person approving the Location Checkout operation.

The 'File Access Info' Tab:

GROUP	FIELD	DESCRIPTION
General	Last Access for Modify Date	Identifies the date on which the document was last opened.
	Last Access for Modify Time	Identifies the time at which the document was last opened.
	Last Accessed	Identifies the date on which the document was last accessed.
	Rename/Copied From	Identifies the previous name of the document, if any
CADD	Total Activity Time	Total of time the document has been opened
	Last Plotted Date	Requires CADNET Plotstation.
	# Times Plotted	Requires CADNET Plotstation
File Info	File Format	Assigned from the File Format/Applications Table.
	Created By	Identifies the user who created the document.
	Created Date	Identifies the date on which the document was created.
	Percent Complete	Assigned by administrator.
	Percent Complete Date	Automatically updated as 'Percent Complete' is modified.
Archive/Back-up Info	Archive Disk	Identifies the Archive Volume.
	Date	Identifies the Archive Date.
	Time	Identifies the Archive Time.
	Backup Disk	Identifies the Backup Volume.
	Date	Identifies the Backup Date.

The 'System Keywords' Tab: See Chapter 9.

The 'Title Block Info' Tab: See Chapter 11.

The 'File Format Keywords' Tab: See Chapter 9.

Exporting to ASCII

The Export To ASCII command creates a delimited ASCII file containing information retrieved from most of the FALCON/DMS database record fields. The Database Export To ASCII dialog box provides an efficient way to build a 'fields to export' list; five field delimiters are available.

To open the Database Export To ASCII dialog box:

From the Administrator, select Database > Export To > ASCII File. The Database Export To ASCII dialog box opens.
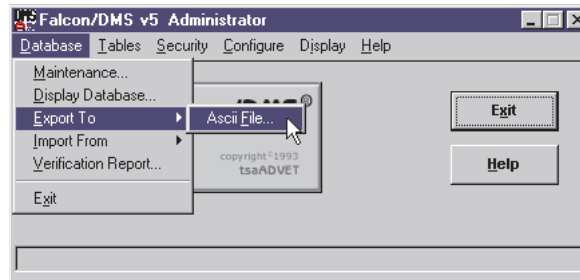
1. Designate an output filename, the default is <falcon_sdir>\log\export.log.

2. From the Fields To Export window, select the fields from which you wish to retrieve information.

3. From the Delimiter drop-down list box, select a Field Delimiter.

4. Press Process to generate the log file.

The log file is a standard delimited and can be read by any application that accepts ASCII format.
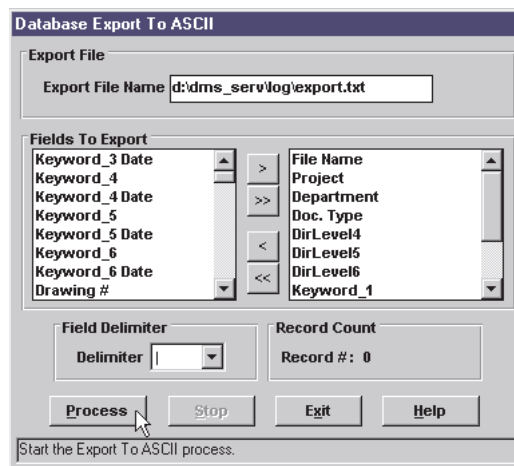
Importing from ASCII

Before implementing true document management, system administrators often use off-the-shelf database programs to maintain basic check in/checkout control over server-based files. There are also organizations, relying on a 'sneaker-net', that build (manually) and maintain (laboriously) a record-keeping system for their digital files. The point is that if your organization implemented some manner of document control, you may already possess a valuable resource that will make your transition to true document management easier and more immediately fruitful. Because Falcon/DMS can import ASCII formatted files, if such a database exists it is likely to contain information that you can use in the Falcon/DMS database.

Before running the Import From ASCII command, you must first edit the ASCII format file exported by the source database. The purpose is to allow you to map delimited items in the ASCII file to specific fields in the Falcon/DMS database record. On the first line of the output file, key in:

RECORD_FORMAT=field#,field#,field#,field#field#,field#,etc

The field#'s are listed in Appendix F.

Basic Rules for Importing from ASCII:

❏ [See Appendix F] The fields identified with an asterisk (★) are required for the database records of digital documents.
❏ Only the Filename (#1) and File Format (#22) fields are required for the database records of paper documents.
❏ Use two consecutive delimiter characters (||) with no space, to skip a field for a record.

To open the Import ASCII dialog box:

From the Administrator, select Database > Import From > ASCII File.

To import an ASCII delimited database file:

1. In the Filename field, enter a filename or press Browse.

2. From the Mode drop-down list box, select an import mode.

   ### The three import Modes are:
   ❏ Update: Updates existing records with metadata contained in the ASCII file.
   ❏ Create: Creates new records from filename, location, file format, and metadata contained in the ASCII file.
   ❏ Update/Create: Existing records are updated. When a specified record is not present, one is created.

3. From the Delimiter drop-down list box, select the delimiter used in the ASCII file.

4. Press Process.
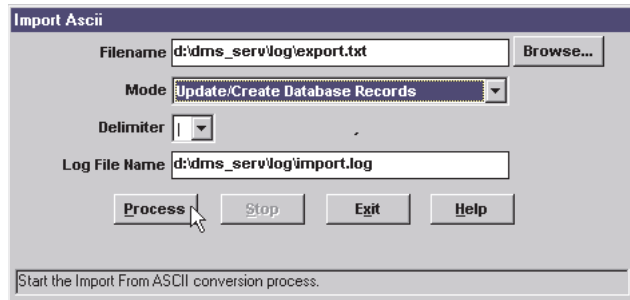
The Database 'Verification Report'

A Database Verification enables an administrator to identify any discrepancies existing between the Falcon/DMS database and data repository. This operation is able to identify both records for which repository-based files are not found, and any repository-based files for which database records do not exist.

Using the Database Verification Report as part of the MassAdd phase of an implementation is a particularly effective way to identify files that have not been added to the database.

To open the Database Verification dialog box

From the Administrator, select Database > Verification. The Database Verification dialog box opens.
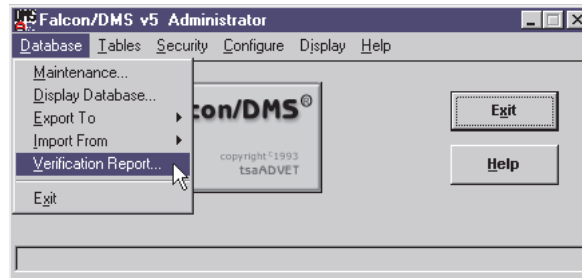
To run Database Verification:

1. Key in a destination log file or accept the default.

2. Enable one or both Report Types:

   ❏ Check for Matching File: When enabled, the verification identifies any database record for which a corresponding data repository file was not found.

   ❏ Check for Matching Record: When enabled, the verification identifies any server-based file for which a corresponding database record was not found.

3. Click the Process button.

Falcon/DMS

The Administrator

Chapter 9:
Database
Keywords
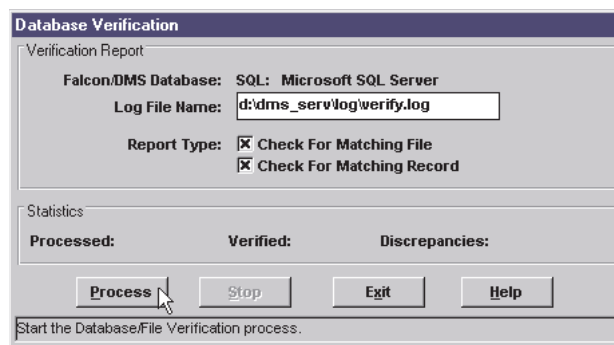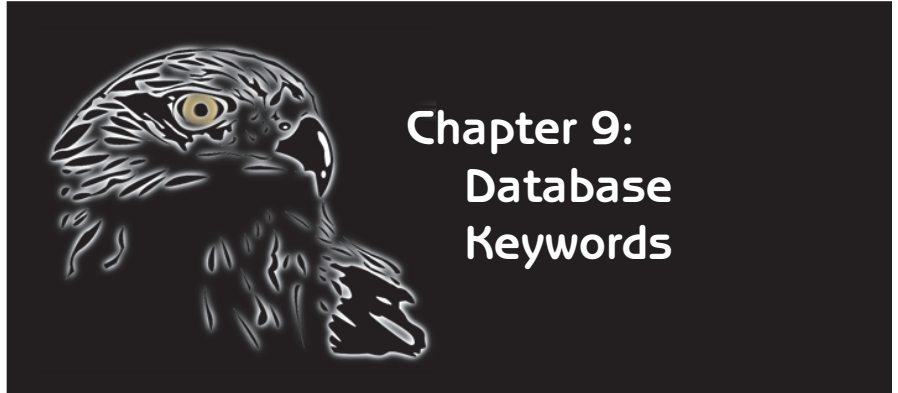
A keyword is an attribute assigned to a database record that facilitates the defining and refining of search criteria.

Keywords are organized by category. Each keyword category is defined by the Falcon/DMS administrator and can be assigned a label that indicates the classification of keywords available within that category.

Falcon/DMS provides two types of keyword categories: system keywords and file format keywords. You can define up to six system keyword categories and up to twenty categories for each Falcon/DMS file format. Within each category you can define an unlimited number of keywords. Additionally, when Multiple Environments is implemented, each environment can have its own set of system and file format keyword categories (see Chapter 7), further expanding the number of keyword categories supported by Falcon/DMS.

Keywords can also be configured to force compliance with a standard keyword scheme. When Keyword Validation is enabled for any single or combination of keyword categories, users are able to assign only those keywords that appear in the associated keyword lists.

This chapter covers:
❏ What the user sees.
❏ Defining system keyword categories and tables.
❏ Defining file format keyword categories and tables.
❏ Keyword validation.
❏ Keywords and multiple environments.

Keywords, What the user sees

Users can view, assign, and search on keywords from both Document Managers. The six system keyword categories are found on the Directories and Keywords tab; the number of keyword categories available depends on how many category labels are defined. Generic labels, keyword1 through keyword6, are predefined for system keywords; in the example below, the first four are customized and the remaining two are not defined.
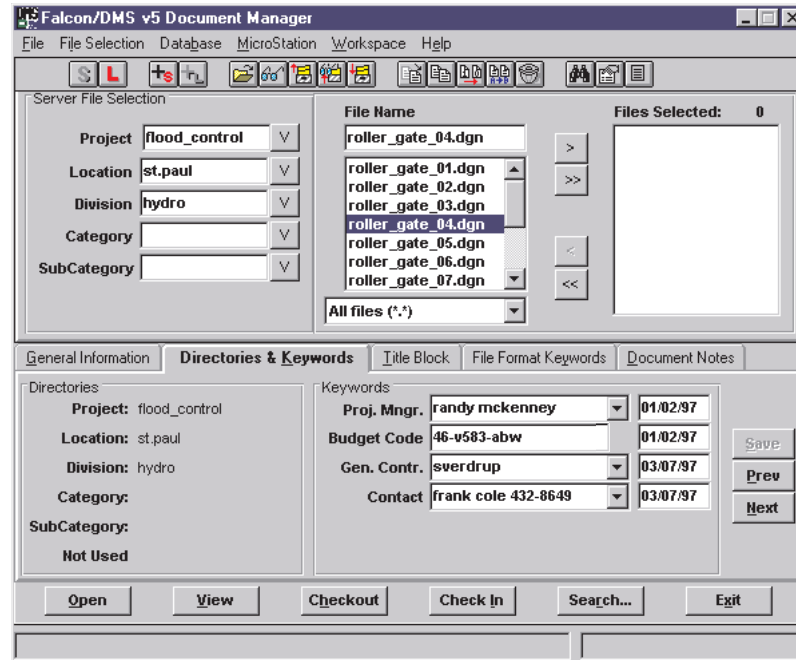
The file format keyword categories are found on the File Format Keywords tab; the number of keyword categories available depends on the file format of the selected document, and how many category labels are defined for that particular file format. A set of generic labels, key1 through key20, are predefined for MicroStation's igds file format; in the example below, the first three labels have been customized.
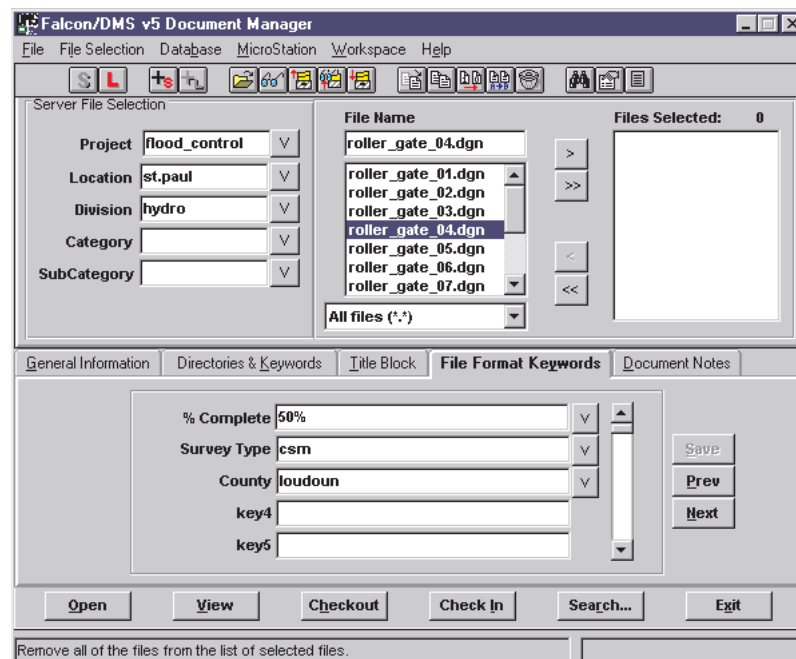
Keywords can be assigned from drop–down list boxes whose contents are defined by the Falcon/DMS administrator. In the example below, a keyword from the Proj. Mngr. category is assigned from a drop–down list box. Predefined lists are supported for both system and file format keywords.
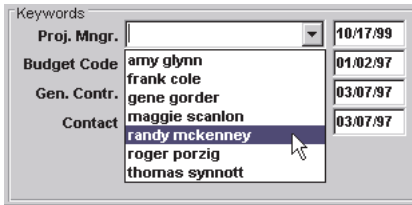
Keywords can be also assigned by entering information in the keyword field. For example, the Budget Code category shown below lists information for which building a predefined list might not be practical. Such input fields are also supported for both system and file format keywords.
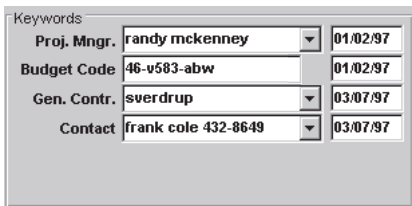
Keyword Validation gives you control over what keywords can be assigned to selected system and file format categories. Shown below is a typical message a user sees when he attempts to enter a keyword that is not in the predefined list. In this example, the keyword ventram has been entered in the Gen. Contr. Category; ventram is not in the predefined list. Keyword Validation is a control that should be applied only to those categories that require users to assign a keyword from a predefined list.
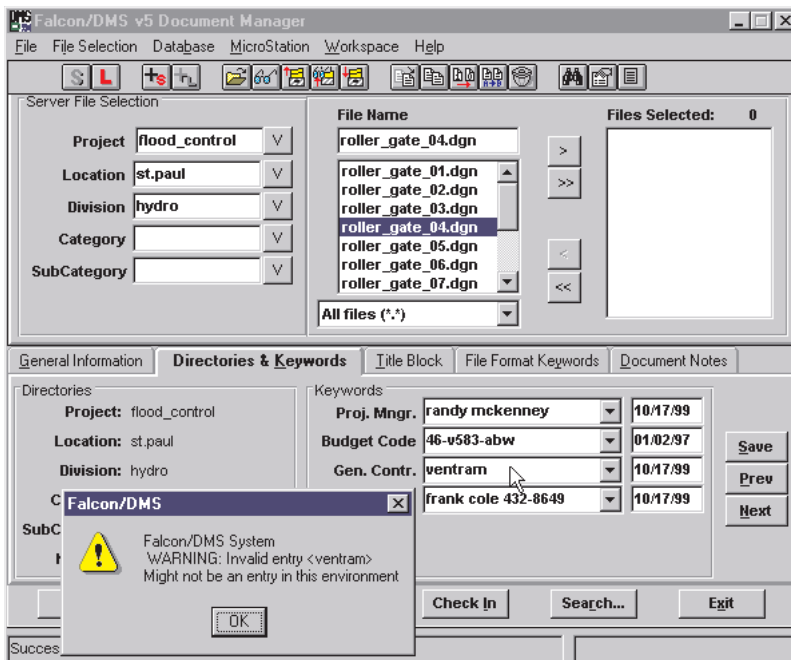
Enabling Keyword Validation for both system and file format keyword categories is discussed on later in this chapter.

The Database Search form allows you to define search criteria for both system and file format keywords. Please note that in the current version of Falcon/DMS, different methods are used to search on system and file format categories.

System Keywords: The system keywords are available from the Directories and Keywords tab on the Database Search form. For system keywords, users define the search criteria by either selecting keywords from predefined lists, or by entering the keyword in the appropriate category.

Database Search

| General Information | **Directories & Keywords** | Title Block | Supplemental |

Directory Levels

Project		V
Location		V
Division		V
Category		V
SubCategory		V

Keywords

Proj. Mngr.	maggie scanlon	▼	Date	
Budget Code		▼	Date	
Gen. Contr.	dravo	▼	Date	
Contact		▼	Date	

Submit SQL   Clear   Exit

Search the database for records that match the selected search criteria.

File Format Keywords: The file format keywords are available from the Supplemental tab on the Database Search form. For file format keywords, users must define the search criteria by first locating the appropriate entry in the DB Fields list and then entering and accepting the desired keyword in the Edit Search Criteria field. In the current version of Falcon/DMS, search criteria for file format categories cannot be defined from predefined lists.

Database Search

| General Information | Directories & Keywords | Title Block | **Supplemental** |

DB Fields   Search Criteria

Share Name	
File Format Keyword 1	
File Format Keyword 2	alta
File Format Keyword 3	

Edit Search Criteria

File Format

alta

Edit   Accept

Submit SQL   Clear   Exit

Search the database for records that match the selected search criteria.

Defining System Keyword Labels & Tables

The system keyword categories are managed from the Tables environment found on the Falcon/DMS Administrator. Falcon/DMS supports six system keyword categories; within each of the six categories you can define an unlimited number of keywords. The keywords are maintained in separate templates, or tables, visible to the user from drop-down list boxes.

To define your own keyword category scheme:

From the Administrator, select Tables > System Labels/Templates > Database Keyword Labels/Templates…
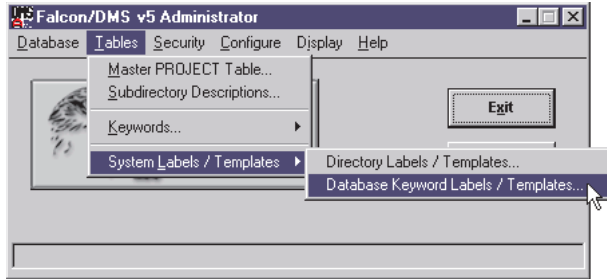
The Database Keyword Labels dialog box opens and displays the six, predefined generic labels, keyword1 through keyword6.
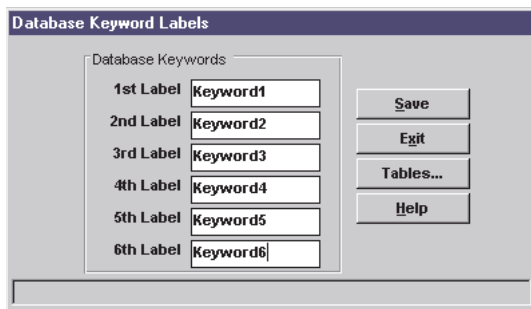
To define a system keyword scheme with four categories:

1. Double-click on keyword1 in the 1st Label field; enter an appropriate label (the maximum length is 12 alpha-numeric characters). Repeat for the 2nd Label, 3rd Label, and 4th Label fields.

2. Double-click on keyword6 in the 6th Label field and press <Delete>, and repeat for the 5th Label field.

   While you can clear label fields to display less than the maximum of six keyword categories, you must begin the suppression of labels from the bottom of the list and work up.

3. Press Save.

   At this point you can press Exit or proceed to define the contents of each keyword category list.

   To define the keywords that will appear in the drop-down list boxes, press Tables...

To populate system keyword tables:

System Keyword Tables

| Contact |
| Proj. Mngr. | Budget Code | Gen. Contr. |

bentram
dravo
eichleay
heery
lockwood greene
septa
sverdrup
u.s.army c.o.e.
ventron

Save
Exit
Add
Remove

Gen. Contr. [   ]

1. On the System Keyword Tables dialog box, select a tab.

2. In the input field located at the bottom of this form, enter a keyword (the maximum length is 40 alpha–numeric characters). Press <Enter> or Add.

3. Enter as many keywords for each category as required. Categories for which it is not practical to define lists should remain blank.

System Keyword Tables

| Contact |
| Proj. Mngr. | **Budget Code** | Gen. Contr. |

Save
Exit
Add
Remove

Budget Code [   ]

4. Press Save and Exit.

To edit a keyword:

1. Double–click a keyword. The selected keyword appears in the input field located at the bottom of the form.

2. Edit as required. Press <Enter>. Repeat as needed.

3. Press Save and Exit.

To remove a keyword or keywords:

1. Select a single keyword or use <Ctrl>+click or press & drag combinations to select two or more keywords.

2. Press Remove. Repeat as needed.

3. Press Save and Exit.

Defining File Format Keyword Labels & Tables

The file format keyword categories are managed from the File Formats/Applications environment found on the Falcon/DMS Administrator. Falcon/DMS supports twenty system keyword categories for each file format; within each category you can define an unlimited number of keywords. The keywords are maintained in separate templates, or tables, visible to the user from drop-down list boxes.

To define your own keyword category scheme:

From the Administrator, select Configure > File Formats/Applications > Edit Global Configuration…
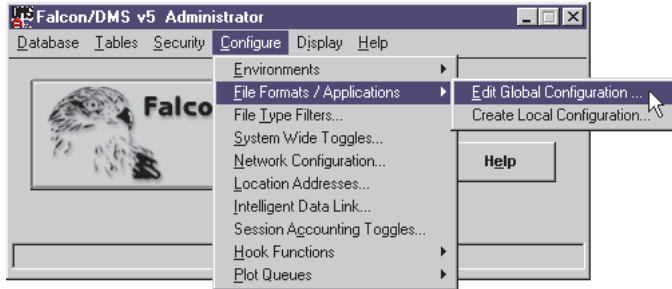
The File Formats/Applications dialog box opens. Because a set of generic labels, key1 through key20, are predefined for MicroStation's igds file format, we will use igds to explain the process. As a model for this explanation we will define the file format keyword scheme used as the example provided in the 'What the user sees' section at the beginning of this chapter.
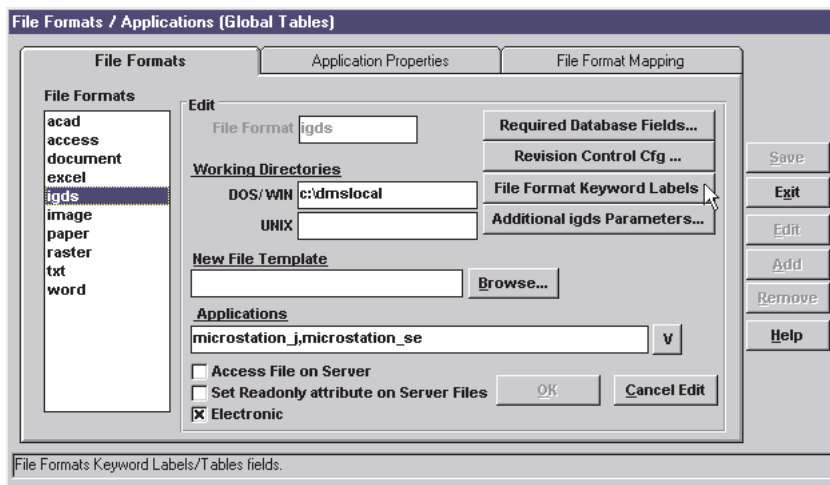
In the File Formats list, double-click igds. Press File Format Keyword Labels.

The File Format Keyword Labels/Tables dialog box opens.

To define a file format keyword scheme with three categories:

1. Double-click on key1 in field 1; enter an appropriate label (the maximum length is 12 alpha–numeric characters). Repeat for the 2nd Label, and 3rd Label.

2. Double-click on key4 in field 4 and press <Delete>; repeat for the remaining fields.

3. Enable the List Exists check boxes for fields 1 through 3. When enabled, these check boxes identify for the Document Managers which tables are available.

4. Press Tables. The File Format Keyword Tables dialog box opens. Follow the procedure explained in the section that describes populating the system keyword tables. When completed, press Save and Exit.

5. On the File Format Keyword Labels/Tables dialog box, press OK.

6. On the File Format/Applications, press OK and Save.

To edit and/or remove a keyword/keywords:

Follow the procedure explained in the sections that describes editing and removing the system keyword tables.

Keyword Validation

Keyword Validation gives you control over what keywords can be assigned to selected system and file format categories. Specifically, for keyword categories for which validation is enabled, users can assign only those keywords that appear in the predefined tables. If a user selects a keyword that is not listed, Falcon/DMS will not save the change to the database record. Do not enable validation for categories for which users can enter their own keywords.

To enable Keyword Validation for system keyword categories:

From the Administrator, select Configure > System Wide Toggles...
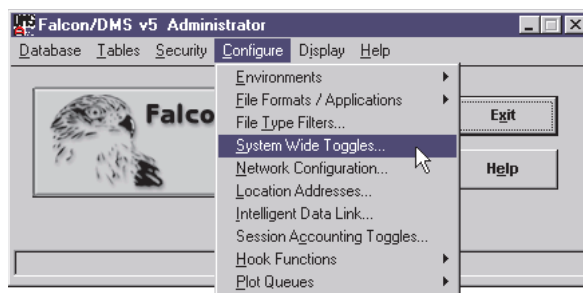
The System Wide Toggles dialog box opens. In the Keyword Validation region, enable the categories for which users will be able to assign only those keywords that are listed in the associated tables. Keyword Validation is a control that should be applied only to those categories that require users to assign a keyword from a predefined list—Budget Code has been identified as a keyword category for which defining a list is not practical.
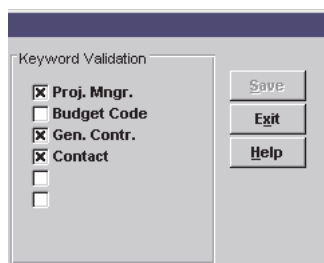
To enable Keyword Validation for file format keyword categories:

On File Format Keyword Labels/Tables dialog box (see previous page), check the Validate check box next to the categories for which users will be able to assign only those keywords that are listed in the associated tables.
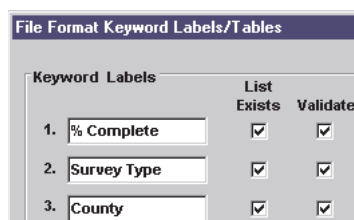
Keywords and Multiple Environments

Multiple keyword schemes for both file format keywords and system keywords can be provided with the Multiple Environments feature (see Chapter 7).

Shown below is the Environment Configuration dialog box, the environment from which you would define multiple keyword schemes. Note that file format keywords and system keywords have separate controls: File format keywords are defined by selecting Specified for the File Formats and Applications category, while system keywords are defined by selecting Specified for the Keyword Labels and Tables category.

Falcon/DMS

The Administrator

Chapter 10:
Location
& Archive
Commands

It is standard practice for a design firm to send documents to a client for approval, or to release a set of documents to a manufacturer for fabrication. To efficiently organize and package files for such purposes, Falcon/DMS provides the Location Checkout command. Location Checkout enables you to send files to an off-site location, while maintaining the data repository version of the same documents under a special status.

For example, as in the cases cited above, when the record versions of documents are in the hands of the client or consultant, it is usually imperative that the data repository versions of these files not be modified, while, at the same, allowing designers to view these files. Location Checkout accomplishes this by automatically rendering these files as view_only documents. There may even be a case in which you will want to move the documents off the data repository entirely; this also can be done as part of the Location Checkout process.

Additionally, when you use Location Checkout to prepare files for delivery to an off-site location, Falcon/DMS can provide the documents in a compressed package for a more efficient transfer.

Also, you can automatically generate a transmittal letter that includes special instructions. If the recipient is another Falcon/DMS user, Location Checkout can provide the database records for the selected documents.

The Location Check In command is used to restore files that have been sent off-site and have now been returned. This process can take files from a consultant or the client who, for example, has made modifications to the designs, update the data repository documents with these more current versions, and update the database as required. The sender might also include new documents with the returning originals. The Location Check In command can automatically add these new documents to the database and move them to the data repository.

The Archive command allows you to select files and move them to a location from which they are then physically archived, using a method of your choosing, while maintaining the archival information in the database. Conversely, Unarchive restores archived files and automatically updates their database records.

The Location & Archive Commands

In this chapter we have to shift our focus from the Falcon/DMS Administrator to the Windows Document Manager. Though the Location and Archive commands are considered to be administrator functions, the Windows Document Manager is used exclusively in the operation of these commands. The single configuration dialog box available from both the Administrator and the Document Manager is the Location Addresses form, used to define the off-site locations.

Let's look first at the Location commands.

The Location Checkout… Command

Location Checkout is used to copy or move documents to an 'outbox'—usually a server directory—from which the files can be easily copied to tape or compact disk. It is important to understand that while Falcon/DMS places files in the outbox directory, it is the administrator who then transfers the files from the outbox to the appropriate media, and then sending the media to the specified location.

To open the Location Checkout dialog box:

After selecting the files in the Windows Document Manager, either as the contents of the current directory or from a search result, choose File > Location Checkout.
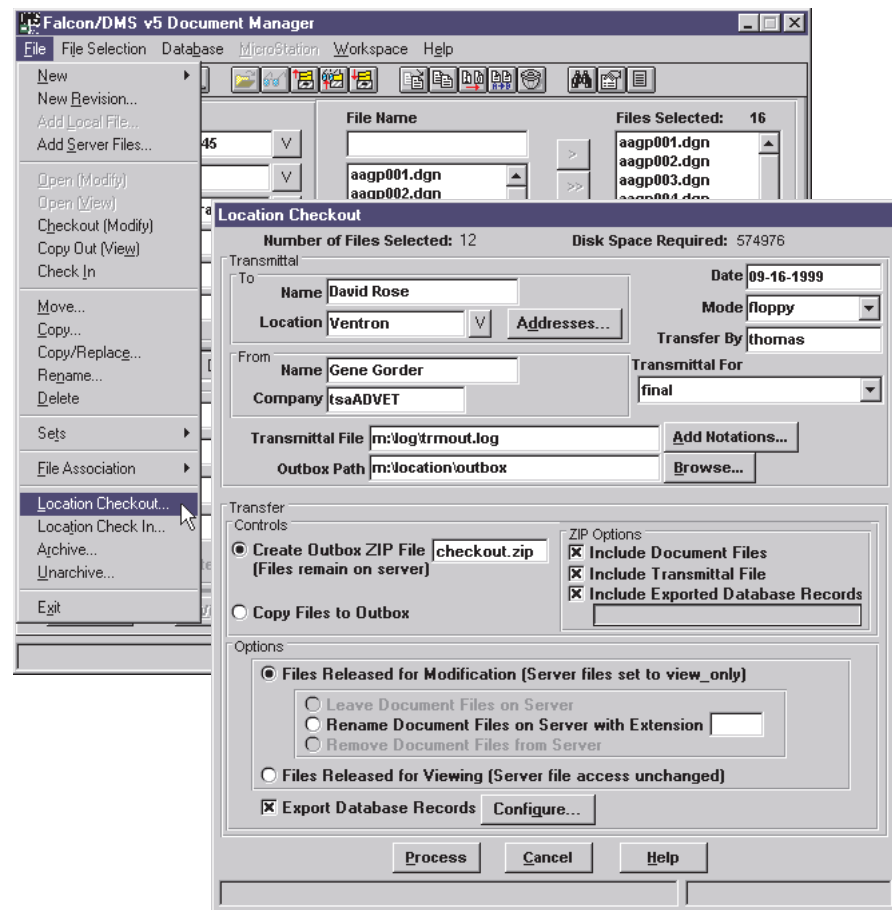
We will talk more about the operation of Location Checkout later in this chapter. First, let's discuss what actually happens with Location Checkout and then examine the primary configuration issues.

Behind the Scenes:

The installation program creates a pair of directories on the application server:

❑ (falcon_sdir)\location\outbox for the Location Checkout command
❑ (falcon_sdir)\location\inbox for the Location Check In command.

You may choose to use these or create your own outbox and inbox directories.

In the case of Location Checkout, the outbox is the target directory into which the files selected for off-site use are placed. Each location, or destination, must identify an outbox directory. (A location is defined from the Location Addresses dialog box and will include the company name and address that appears on the transmittal letter; the Location Addresses dialog box is explained later in this chapter.)

After selecting the documents intended for off-site use and launching the Location Checkout command, Falcon/DMS moves or copies the files to the outbox used by the designated location. It is from this directory that you will place the files on the appropriate media and deliver them to the off-site user.

Primary Configuration Issues:

The Location Checkout command transfers files to an outbox in one of two modes. These modes are referred to as Transfer Controls:

Transfer Controls:

❑ **Create Outbox Zip File:** When chosen, Location Checkout never removes the selected documents from the data repository. However, Falcon/DMS can automatically rename the data repository versions of the documents by assigning the default extension, '.cop', or one that you define. There is a significant benefit to using the default extension: when '.cop' is used to rename files, the database records for the original documents are assigned to the renamed files. When '.cop' is not used the renamed versions of the 'At Location' documents do not have database records and will not be visible when the Document Manager's Server View Mode is set to 'Database'.

❑ **Copy Files to Outbox:** You have the option of leaving a record version of the selected documents on the data repository or removing the documents when this control is selected. You can also choose to automatically rename the data repository versions of the documents by assigning a new filename extension.

Transfer Options:

The Transfer Controls have two Transfer Options:

❑ Files Released for Modification (Server files set to view_only).
❑ Files Released for Viewing (Server file access unchanged).

The second option, 'Files Released for Viewing (Server file access unchanged)', does not alter the Availability of the selected documents. When chosen, Location Checkout simply packages the selected documents as specified in the Transfer Controls section and places the documents in the location's outbox directory.

The first option, 'Files Released for Modification (Server files set to view_only), is where the real power of Location Checkout is apparent.

Transfer Modes for Option 2—Files Released for Modification:

It should be noted that the three modes in which this option functions are available only when the second Transfer Control, Copy Files to Outbox, is selected. (When Create Outbox Zip file is chosen only the second mode is available.)

The three modes are:

❑ **Leave Document Files on Server:** When selected, Location Checkout copies the selected files to the outbox and sets the Availability for the files on the data repository to 'At Location'. The database also lists the location and the date of the transaction for each file.

❑ **Rename Document Files on Servers with Extension:** When selected, Location Check-out copies the selected files to the outbox, sets the Availability for the relevant database records to 'At Location', and renames the files on the data repository with the specified extension. Unless the default extension, '.cop' is used, the renamed files do not have database records.

❑ **Remove Document Files from Server:** When selected, Location Checkout moves the selected files to the outbox. The database records for the affected files remain and are visible when the Server View Mode is set to database. The Availability for each document is 'At Location'.

Special Considerations

When documents are sent off-site, the Location Checkout command can be configured to automatically rename the data repository versions of the documents with a user defined extension. When you select the first transfer control—Files Released for Modification (Server files set to view_only)—and set the option to 'Rename Document Files on Servers with Extension', a default extension displays in the option's input field; this default extension is '.cop'. When '.cop' is used, the data repository versions of the documents are associated with the original database records. When you assign a different extension the renamed files do not have database records. (You might use a different extension when you want the extension to help identify the consultant to whom the files were sent) Using a different extension creates a situation that requires special consideration for Document Manager users in general and MicroStation users in particular:

❑ As indicated, using an extension other than '.cop' to rename data repository files produces files that do not have database records. This creates a problem when users require view access to the data repository version of an 'At Location' document. To view a document through Falcon/DMS, the document must have a database record. However, for 'At Location' documents any new database record must somehow indicate that the associated document cannot be modified. This is done most efficiently by defin-ing a Document Status Access that restricts access to view_only and applying this status as you create database records for the renamed files, usually with the Add Server command. When the original files are restored with Location Check In, you can easily delete the renamed files. This, of course, means that the database records for the original files are no longer available. To account for this, when the Location Checkout is performed, export the database records for the off-site files. When the documents are restored, import the database records with the Import Mode set to 'Create database records'.

❑ When MicroStation files are sent off-site and their repository versions are assigned a new extension, it is important to consider what impact renaming these files will have on reference file attachments.

The Location Checkout dialog box:

GROUP	OPTION	DESCRIPTION
Statistics	Number of files selected:	Displays the number of files selected.
	Disk Space Required:	Displays the amount of disk space the selected files will require on the media used to transfer the documents from the outbox directory to the off-site location.
Transmittal	To:	[Name] Enter the Recipient's name as it will appear on the transmittal letter. [Location] Press **V** to open and choose from the Existing Locations list. [Note: The location also populates the Outbox Path field, below.]
	Addresses...	Press to open the Location Addresses dialog box. This form is used to add to, delete from, and modify entries in the Existing Locations list. The Location Addresses dialog box is also available from the Administrator [select Configure > Location Addresses].
	From	Name & Location: Enter sender's name and organization.
	Date	Date processed.
	Mode	Indicates the media used to transfer the selected files from the outbox directory to the off-site location.
	Transfer By	Defaults to the current login account name.
	Transmittal For	Indicates the purpose of the Location Checkout transaction. The contents of the drop-down list box items are user defined. The list is built in an ASCII text file -- /falcon_sdir/cfg/transfor.cfg.
	Transmittal File	Indicates the location and name of the log file containing the transmittal information [edit as required].
	Add Notations	Opens the Transmittal Notations dialog box from which you can specify notes and special instructions to appear on the transmittal letter.
	Outbox Path	This indicates the location to which Location Checkout copies or moves the selected files and is listed when a Location is selected [see 'To:', above]. Each Location must have a valid outbox path -- a valid outbox path is an existing directory.
	Browse...	Press Browse to select a directory other than the specified outbox path as defined in the Location Addresses dialog box.
Transfer Controls	Create Outbox Zip File -- Files remain on server	When selected, enter a filename -- the zip file is created automatically. The Zip Options, listed below, are available only when Create Oubox Zip File is selected.
		Zip Options [available only with Create Outbox Zip File option]:
		Include document files: When enabled, this option will compress the selected files into the the specfied zip file.
		TIP: You can use the Location Checkout command as a way to export the database records of selected files in much the same that you export the entire database using the Database > Export to ASCII command. Simply disable this option, 'Include document files', and enable the 'Include exported database records' option, below.
		Include transmittal file: When enabled, this option includes the transmittal file in the specified zip file.
		Include exported database records: When enabled, this option includes the database records for the selected files. This option is available only if the Export Database Records option, below, is enabled.
	Copy Files to Outbox	Allows you to choose between copying the selected files to the outbox and leaving a view_only version on the data repository, assigning a new extension, or moving the files off the data repository and into the outbox. This option does not permit you to automatically compress the selected files through Location Checkout.
	Note:	The Options > Files Released for Modification controls listed below apply to theTransfer Controls radio button options: -- The Create Outbox Zip File control has access to only the Rename document files on server option. -- The Copy Files to Outbox control has access to all three options.

GROUP	OPTION	DESCRIPTION
Options	Files Released for Modification -- Server files set to view_only	Leave document files on server: Copies the selected files to the outbox; the Availability for the relevant files on the data repository is set to 'At Location'.
		Rename document files on server with extension: Copies the selected files to the outbox; the Availability for the relevant database records is set to 'At Location'; the files are renamed with the specified extension -- these files do not have database records.
		Remove document files from server: Copies the selected files to the outbox; deletes the relevant files from the server; the database records remain with the Availability for each set to 'At Location'.
	Files Released for Viewing -- Server files unchanged	When enabled, the selected files are simply copied to the outbox; no changes are made to the files' database records.
	Export Database Records	When enabled, the database records for the selected files are written to an ASCII text file and moved to the outbox.
	Configure...	Opens the Configure Export Fields dialog box. This form allows you to specifiy the name of the export file and identify the database fields to be exported.

The Location Addresses dialog box:

The Location Addresses dialog box is used to define a Location and its parameters. This form is available from both the Administrator and the Windows Document Manager.

To open the Location Addresses dialog box:

From the Administrator, select Configure > Location Addresses.
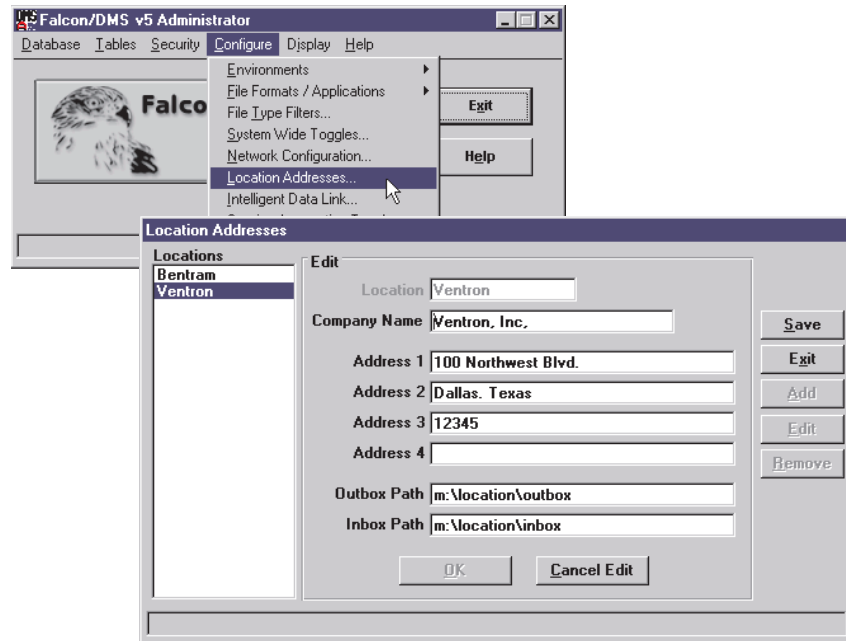
To open the Location Addresses dialog box from the Document Manager, select File > Location Checkout. On the Location Checkout form, press Addresses.
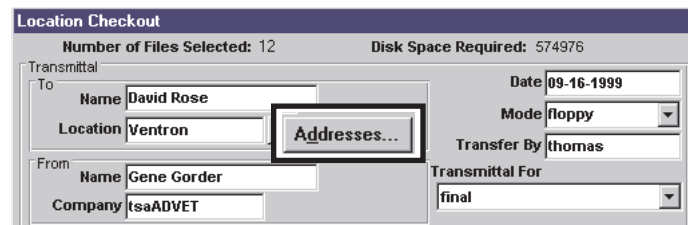
Location address parameters include the Location name, the name of the organization associated with that location, address information, and the outbox and inbox paths. The organization and address information appears as entered on the transmittal letter automatically generated by the Location Checkout command. The outbox path identifies the directory to which the Location Checkout command transfers documents for delivery off-site; the inbox path identifies the location from which the Location Check In command restores the documents when they are returned from the off-site location. It is not necessary for the actual directories to exist at the time the Location Address is defined. However, the physical directories must exist, as defined, when you use either of the Location commands to checkout or check in documents. It is not necessary to assign a server-based directory as an outbox or inbox; pointing the outbox to a directory on the administrator's workstation is often more convenient than copying or moving files to a server-based outbox.

To send documents to an off-site location [Location Checkout]:

1. From the Document Manager, select the files you wish to send to an off-site location. The selection can be defined either as the contents of the current directory or from a search result,

2. From the Document Manager, choose File > Location Checkout. The Location Checkout dialog box opens.

3. On the Location Checkout dialog box, enter a recipient's name and press [V] in the Transmittal > To region.

   From the Existing Locations list box, select a location. The location must have a valid outbox; to be valid, an outbox must have a corresponding directory that is accessible from the current workstation.
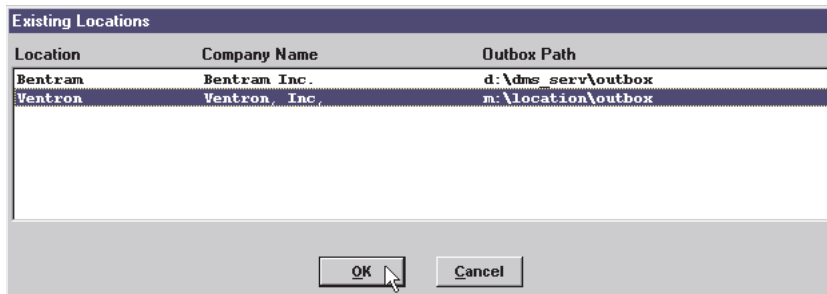
Existing Locations		
Location	**Company Name**	**Outbox Path**
Bentram	Bentram Inc.	d:\dms_serv\outbox
Ventron	Ventron, Inc.	m:\location\outbox

   If the required location is not listed, press Addresses... and define a new Location on the Location Addresses dialog box.

   Define the contents of the remaining Transmittal fields as required. Press Add Notations... to define any special information that you wish to include on the Transmittal. Press Browse... to select a different outbox directory.

4. In the Transfer Controls and Transfer Options regions, establish the required parameters.

   To provide database records for the selected files, enable 'Export Database Records'. To select the database fields you wish to export, press Configure...

   Select the field you wish to export. Press OK.

5. On the Location Checkout dialog box, press Process.

The Location Check In... Command

The Location Check In command is used to restore files that have been sent off-site and have now been returned and placed in an 'inbox'. An inbox is the directory into which files returned from a location are placed. The Location Check In command then retrieves the files from the inbox and restores the documents to either their home directories or a new 'target' directory on the data repository and updates each file's database record. The sender might also include new documents with the returned originals. The Location Check In command automatically adds new documents to the database while moving them to a target directory on the data repository.

To open the Location Check In dialog box:

After placing the files to be restored in the appropriate inbox directory, from the Windows Document Manager, choose File > Location Check In.
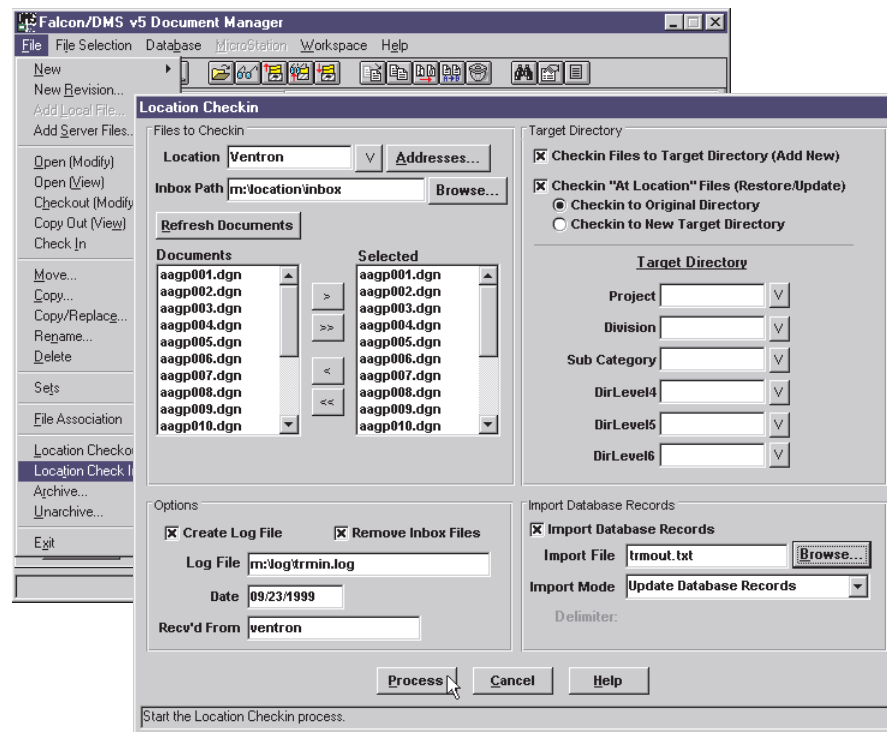
Behind the Scenes:

The Location Check In command automatically assigns an 'igds' File Format to *.dgn files and an 'acad' File Format to *.dwg files. If files with different extensions are selected, the Location Check In command prompts the user for a File Format each time it encounters a different extension—however, the command will remember which File Format to assign the next time it encounters that extension.

Two options are provided in the Target Directory region:

❑ Check in Files to Target Directory (Add New): This option is used to add files and, if provided, their database records to a target location's database, when that target location is a Falcon/DMS site.

❑ Check in 'At Location' Files (Restore/Update): This option is used when files are returned to their originating location. This feature permits the administrator to update the server-based files or place the returning files in a different target directory, while changing the 'availability' of the files from 'at location' to 'on server'.

The Location Check In dialog box:

GROUP	OPTION	DESCRIPTION
Files to Check In	Location	Displays the 'location' from which files are to be returned to the data repository.
	Addresses...	Press to open the Location Addresses dialog box. This form is used to add to, delete from, and modify entries in the Existing Locations list. The Location Addresses dialog box is also available from the Administrator [select Configure > Location Addresses].
	Inbox Path	Indicates the directory from which Location Check In restores files.
	Browse...	Press Browse to select a directory other than the specified inbox path as defined in the Location Addresses dialog box.
	Refresh Documents	Press Refresh Documents to update the Documents list.
Options	Create Log File	When enabled, creates a logfile with the specified parameters.
		The following information fields are editable only when the Create Log File option is set, either as the sole option or in combinatin with 'Remove Inbox Files'.
		Log File: Displays log filename.
		Date: Displays transaction date
		Recv'd From: Displays sender's [Location] name
	Remove Inbox Files	Deletes the files from the inbox upon completion of Location Check In transaction.
Target Directory	Check In Files to Target Directory [Add New]	Restores files from a sender [Location] -- requires that a Target Directory [below] be defined. Capable of creating database records for new files as they are moved to the data repository.
	Check In 'At Location' Files [Restore/Update]	Restores files from a sender [Location]. Capable of creating database records for new files as they are moved to the data repository.
		Check In Options
		Check In to Original Directory: When selected, each document's database record identifies the target directory. [Primary Index Key must be set to 'Filename'.]
		Check In to New Target Directory: The destination directory is determined by the Target Directory setting.
	Target Directory	Use to set destination (Target) directory.
Import Database Records	Import Database Records	When enabled, imports updated database records for selected files.
		The following parameters are available only when Import Database Records is enabled.
		Import File: Displays the database import file
		Browse... Press Browse to assign database import file
		Import Mode: Select one -- Update Database Records: updates existing records -- Create Database Records: creates new database record [matching file not required]. -- Update/Create Database Records: updates existing records and creates new database record
		Delimiter: displays special character used to seperate columns

To restore documents from an off-site location [Location Check In]:

1. From the operating system, place the files to be restored in a directory accessible from the current workstation—this directory is the inbox.

2. From the Document Manager, choose File > Location Check In. The Location Check In dialog box opens.

3. On the Location Check In dialog box, press [V] in the Files to Check In region.

   From the Existing Locations list box, select a location. The location must have a valid inbox; to be valid, an inbox must have a corresponding directory that is accessible from the current workstation.

   If the required location is not listed, press Addresses... and define a new Location on the Location Addresses dialog box.

   Press Browse... to select a different outbox directory.

4. Select the documents you wish to restore.

5. In the Options, Target Directory, and Import Database regions, set the required options (see 'The Location Check In dialog box', preceding page). Press Process.

The Archive... Command:

The Archive command is used to move selected files to a location from which they are archived using a method of your choosing. The date of the archive and the directory to which the documents were archived are captured by Falcon/DMS and written to their database records. Ideally, the selected files are moved to a directory from which the network's archive operations automatically retrieve files.

To open the Archive Documents dialog box:

After selecting the files in the Windows Document Manager, either as the contents of the current directory or from a search result, choose File > Archive.
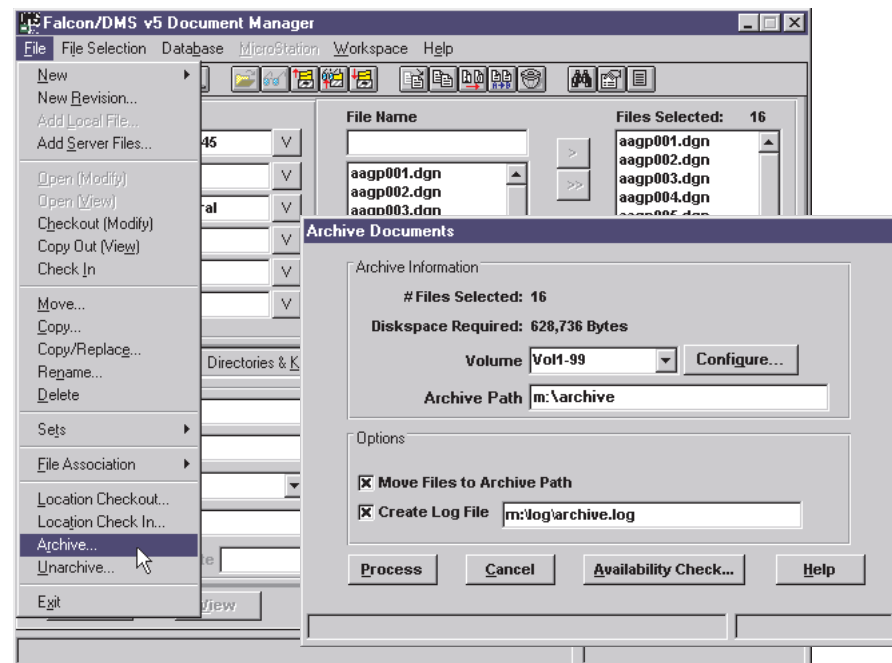
Behind the Scenes:

The Archive Path, the target directory from which the administrator retrieves the files for the actual archival, does not have to exist. The Archive command will create the directory if it is not found.

To Archive documents:

1. From the Document Manager, select the files you wish to archive. The selection can be defined either as the contents of the current directory or from a search result.

2. From the Document Manager, choose File > Archive. The Archive dialog box opens.

3. In the Archive Information region, select a volume from the Volume drop-down list box. This volume is assigned to each document's database record and helps identify the location of the archived data.

   If the required volume does not exist, Press Configure... and add a volume entry on the Configure Archive Volumes dialog box.

   The Archive Path field displays the target directory to which the selected files are moved; this value is assigned by the selected Volume. It is from this directory that the administrator retrieves the files for the actual archive; if the directory does not exist, the Archive command will create it.

4. In the Options region, select the required options.

   Move Files to Archive Path: This option moves the selected files from their data repository home directory to the Archive Path.

   Create Log File: This option creates a log file; the field is editable.

5. Press Availability Check... to display statistics on the selected files. Specifically, the Archive Availability Check identifies any files whose Availability is not set to 'On Server'; such files will not be archived.
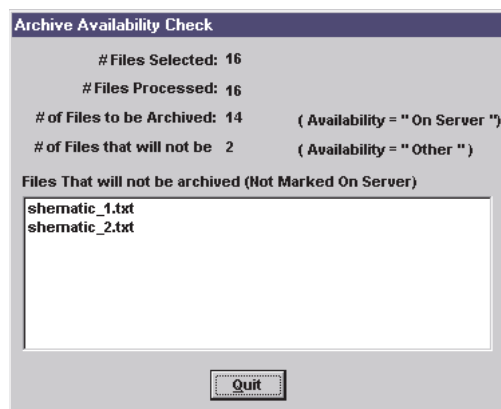
   On the Archive Availability Check box, press Quit.

6. Press Process.

The Unarchive... Command:

The Unarchive command is used to restore archived files to a project's data repository home directory and reset the documents Availability to 'On Server'.

To Unarchive documents:

1. From the operating system, place the files to be unarchived in a directory accessible from the current workstation.

2. From the Document Manager, choose File > Archive. The Archive dialog box opens.
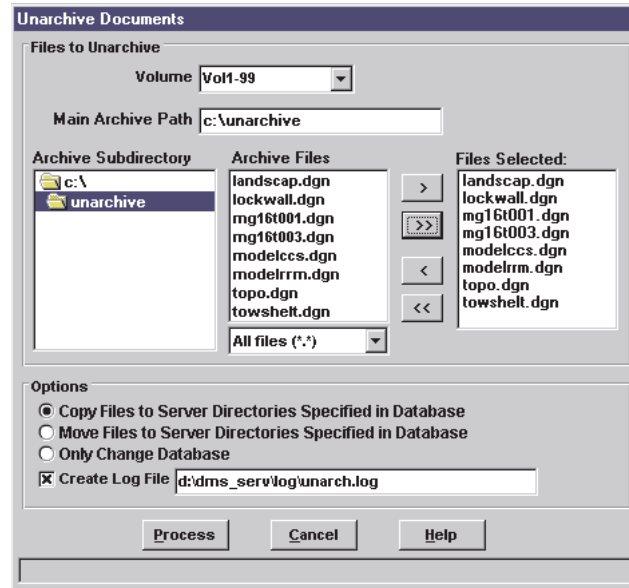
3. In the Files to Unarchive Information region, select a volume from the Volume drop-down list box. The default Main Archive Path is set based on the selected Volume. You can change the path as required.

4. Select the documents you wish to restore.

5. In the Options region, select the required options.

   Copy Files to... : Copies the selected files their original data repository home directories.

   Move Files to... : Moves the selected files their original data repository home directories

   Only Change Database: Changes only the database records of the selected files.

   Create Log File: This option creates a log file; you can specify the output file's filename.

6. Press Process.

Falcon/DMS

The Administrator

Chapter 11:
MicroStation &
AutoCAD
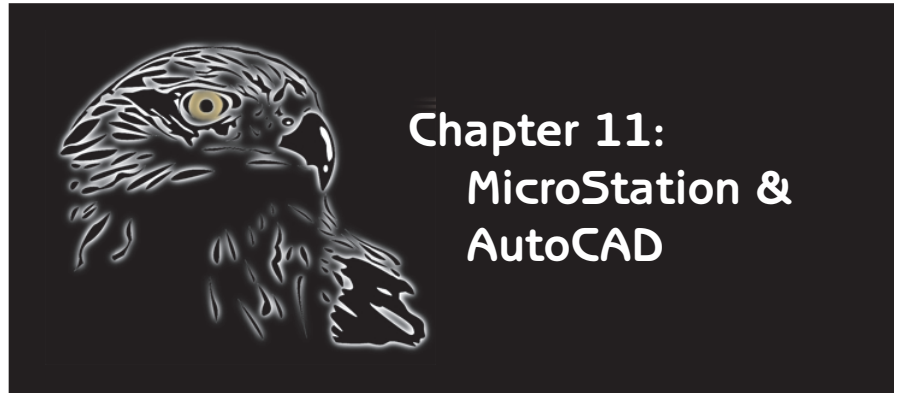
The focus of this chapter is on the special Administrator requirements for MicroStation and AutoCAD, reference file issues for MicroStation, and Intelligent Data Link (MicroStation only).

Specifically, special Administrator requirements refers to the 'Additional IGDS Parameters' and 'Additional ACAD Parameters' forms for MicroStation and AutoCAD, respectively. These setup options are found on the File Formats and Applications dialog box.

Intelligent Data Link, or IDL, refers to the bi-directional link between information presented in the title block of a MicroStation Design file and the file's database record. Falcon/DMS is capable of maintaining this link so that changes made to certain information in the design file will cause the associated fields in the database to update, and vice versa.

The CAD File Formats

File Formats for both MicroStation and AutoCAD are pre configured and delivered as part of the Falcon/DMS installation. Because the general subject of File Formats is presented in detail in Chapter 3, this discussion of the CAD File Formats focuses on issues specific to these applications, with special attention given to the Additional 'CAD' Parameters environments. We will begin this section with the 'Additional igds Parameters' for MicroStation and conclude with an explanation of the 'Additional acad Parameters' for AutoCAD.

Additional igds Parameters—MicroStation
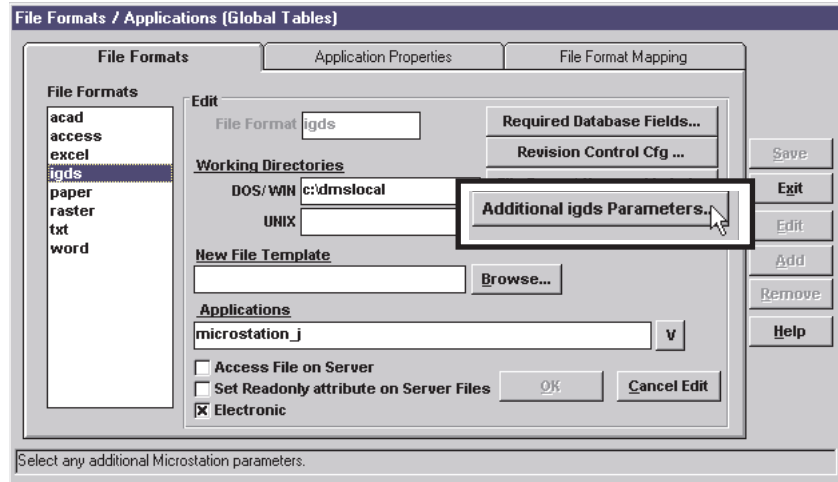
To open the MicroStation Configuration dialog box:

From the Administrator, select Configure > File Formats/Applications. The File Formats/Applications dialog box opens.

In the File Formats list, double-click on igds. Press Additional igds Parameters. The MicroStation Configuration dialog box opens.

Additional igds Parameters

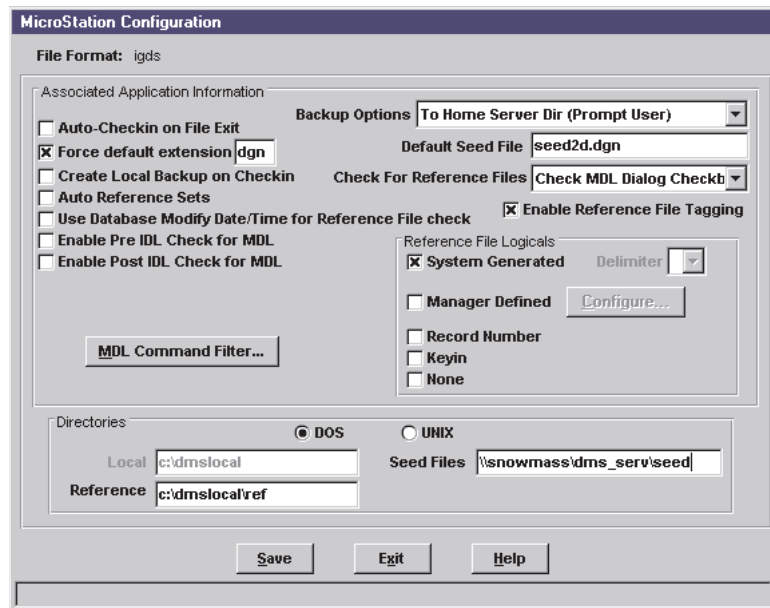| GROUP	OPTION	DESCRIPTION
Associated Application Information	Auto-Check In on File Exit	Automatically checks in a design file as a user either opens a second file or exits MicroStation. While a user can still checkout multiple files, each file that is opened will be automatically checked in.
	Force default extension	Appends the designated extension for the Create and Rename operations.
	Create Local Backup on Check In	Automatically creates a local backup of each file at Check In. Each user will be responsible for maintaining the backup.
	Auto Reference Sets	Automatically checks out all the documents in a set.
	Use Database Modify Date/Time for Reference File check	Uses the database, instead of a directory and file based size-date check, when determining the need to refresh local reference files.
	Enable Pre IDL Check - MDL	Checks for discrepancies between IDL data and a document's record prior to opening a design file.
	Enable Post IDL Check - MDL	Checks for discrepancies between IDL data and a document's record prior to checking in a design file.
	Backup Options	No Backup
		To Home Server Directory (Prompt User): Allows the user to refresh the repository-based record version of document A as the user opens document B, or for the active file as the user exits MicroStation.
		To Home Server Directory (Automatic): Automatically refresh the repository-based record version of document A as the user opens document B, or for the active file as the user exits MicroStation.
		To Server Backup Directory (Prompt User): Allows the user to place a copy of document A in a backup directory under document A's current repository directory as the user opens document B, or for the active file as the user exits MicroStation.
		To Server Backup Directory (Automatic): Automatically places a copy of document A in a backup directory under document A's current repository directory as the user opens document B, or for the active file as the user exits MicroStation.
	Default Seed File	Designates the default seed file.
	Check For Reference Files	Check on Open: Automatically checks for reference files as each design file is opened or checked out.
		Check on Checkout: Automatically checks for reference files as each design file is checked out.
		Check MDL Dialog Checkbox: Determines status of the 'Check For Reference Files' control on the MDL Document Manager when opening or checking out design files. This option gives users the choice of checking for reference files on either open or checkout. A significant benefit of this option is realized when checking out a large number of design files. By unchecking the Document Manager control the user can delay checking for reference files until each document is opened.
		No Check
	Enable Reference File Tagging	Allows users to select the reference files to be copied out.
	Reference File Logicals (Path)	System Generated: Automatically generates reference path logical.
		Manager Defined: Highest precedence--uses a predefined table of path logicals when attaching reference files. If a predefined logical does not exist for an attachment Falcon/DMS assigns a system generated logical.
		Record Number: Identifies each reference file with a unique database identifier as it is attached to the active file. This attribute enables Falcon/DMS to resolve reference file attachments for reference files that have been renamed or moved.
		Keyin: Allows the user to define the path logical.
		None: Not recommended for databases with a Primary Index Key of 'DirLevel1;Filename' and greater.

Additional igds Parameters (cont.)

GROUP	OPTION	DESCRIPTION
Associated Application Information	MDL Command Filter	Open File: Redirects MicroStation's File>Open command to Falcon/DMS. Do not enable if you wish to allow users to open MicroStation design files on their local hard disks.
		Attach Reference File: Redirects MicroStation's Reference>Tools>Attach command to Falcon/DMS. Strongly recommended for databases with a Primary Index Key of 'DirLevel1;Filename' and greater.
		New File: Redirects MicroStation's File>New command to Falcon/DMS
		RD (retrieve design file): Check to disable. Not recommended when using the MDL and Windows Document Managers interchangeably.
		XD (exchange design files): Check to disable.
		Save As: Check to disable.
Directories	DOS/UNIX	Allows you to set directory parameters for the selected OS
	Local	Reads the working directory variable on the File Formats tab.
	Reference	Sets the local 'copy-out' directory for reference files.
	Seed Files	Sets the seed file location.

Additional acad Parameters—AutoCAD

To open the AutoCAD Configuration dialog box:

From the Administrator, select Configure > File Formats/Applications. The File Formats/Applications dialog box opens.
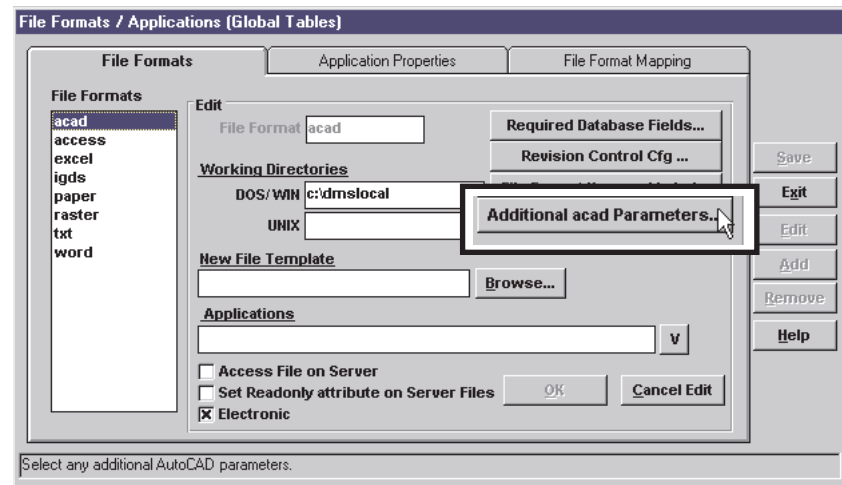
In the File Formats list, double-click on acad. Press Additional acad Parameters. The AutoCAD Configuration dialog box opens.

```
┌────────────────────────────────────────────────────────────────┐
│ AutoCAD Configuration                                            │
│   File Format: acad                                             │
│ ┌─ Associated Application Information ─────────────────────────┐ │
│ │  □ Auto-Checkin on File Exit      Default Prototype File [acad.dwg    ]│
│ │  □ Create Local Backup on Checkin Backup Options [No Backup      ▼]│
│ │                                   Check For XRef Files [Check on Open ▼]│
│ └──────────────────────────────────────────────────────────────┘ │
│ ┌─ Directories ───────── ◉ DOS    ○ UNIX ──────────────────────┐ │
│ │   Local [c:\dmslocal        ]  Prototype Files [\\star1\dms_serv|template\acad]│
│ │ Reference [c:\dmslocal\xref  ]                               │ │
│ └──────────────────────────────────────────────────────────────┘ │
│            [ Save ]     [ Exit ]     [ Help ]                     │
│ [                                                              ] │
└────────────────────────────────────────────────────────────────┘
```

Additional acad Parameters

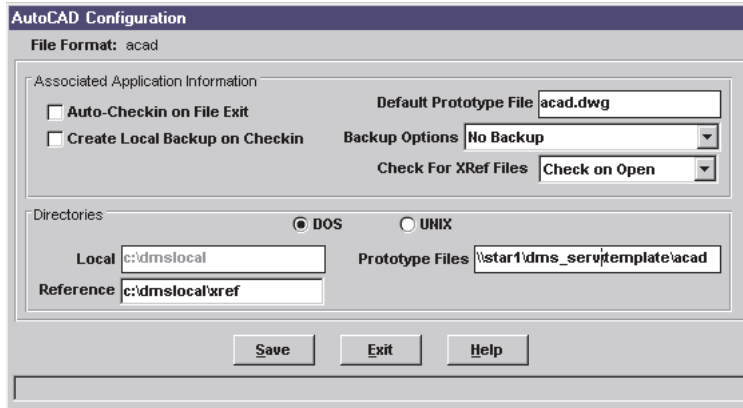| GROUP	OPTION	DESCRIPTION
Associated Application Information	Auto-Check In on File Exit	Automatically checks in a design file as a user either opens a second file or exits MicroStation. While a user can still checkout multiple files, each file that is opened will be automatically checked in.
	Create Local Backup on Check In	Automatically creates a local backup of each file at Check In. Each user will be responsible for maintaining the backup.
	Default Seed File	Designates the default seed file.
	Backup Options	No Backup
		To Home Server Directory (Prompt User): Allows the user to refresh the repository-based record version of document A as the user opens document B, or for the active file as the user exits MicroStation.
		To Home Server Directory (Automatic): Automatically refresh the repository-based record version of document A as the user opens document B, or for the active file as the user exits MicroStation.
		To Server Backup Directory (Prompt User): Allows the user to place a copy of document A in a backup directory under document A's current repository directory as the user opens document B, or for the active file as the user exits MicroStation.
		To Server Backup Directory (Automatic): Automatically places a copy of document A in a backup directory under document A's current repository directory as the user opens document B, or for the active file as the user exits MicroStation.
	Check For XRef Files	Check on Open: Automatically checks for reference files as each design file is opened or checked out.
		Check on Checkout: Automatically checks for reference files as each design file is checked out.
		No Check
Directories	DOS/UNIX	Allows you to set directory parameters for the selected OS
	Local	Reads the working directory variable on the File Formats tab.
	Reference	Sets the local 'copy-out' directory for XRef files.
	Prototype Files	Sets the prototype file location.

Falcon/DMS & Reference Files

The Fundamentals

Reference File Configuration

When a design file to which reference files are attached is opened or checked out, Falcon/DMS copies the reference files as view_only files to the client workstation. (The setupwin program places FALCON.CFG in MicroStation's \config\appl subdirectory. This file sets the MS_RFDIR environment variable to point to (c):\dgn\ref. You may edit this as required.)

There are two MicroStation Configuration options that control how Falcon/DMS will handle a design file to which reference files are attached:

❏ Check for Reference Files

❏ Enable Reference File Tagging
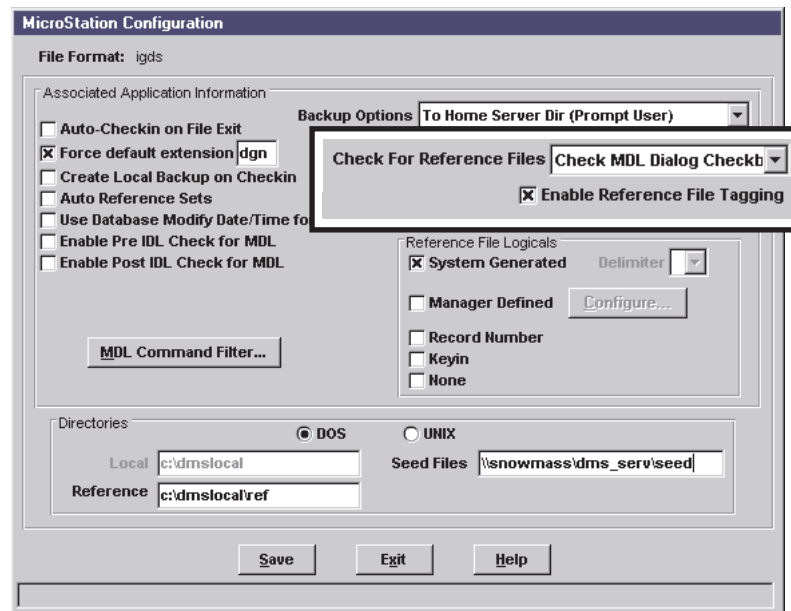
Check for Reference Files:

Determines when Falcon/DMS performs its reference file check.

❏ Check on Open: Enables Falcon/DMS to perform a reference file check each time a user performs the Open or Check Out operations. The reference file check identifies which files, if any, are already copied out to the user's workstation. If files are present, the local files are compared to the data-repository files. When the repository version of a file is more current than the local version, Falcon/DMS refreshes the local version. The verification also determines whether a reference file is checked to another user. When one or more reference files are checked out to another user, Falcon/DMS identifies the person to whom a file is checked out and the workstation.

❏ Check on Checkout: Enables Falcon/DMS to perform the reference file checks explained above, but only for the Check Out operation.

❏ Check MDL Dialog Box: This option enables the user to control when Falcon/DMS checks for reference files by placing a Reference File Check control on the MDL Document Manager.

Open	View	Checkout (Modify)	Copy Out (View)

☐ Unload Document Manager   ☑ Reference File Check

Status: Retrieved record for aagp001.dgn
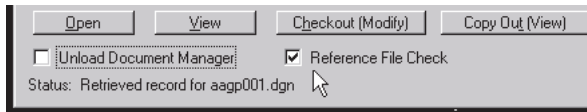
❏ No Check: Enable this option when files are not checked out to the workstation—that is, when files are accessed directly on the server (data repository)—or when reference files are not used.

Enable Reference File Tagging:

Allows users to choose which reference file or files are to be copied to the local workstation.

Falcon / DMS - Reference File Selection

File Name: c:\dmslocal\flood_control\lock10-a.dgn
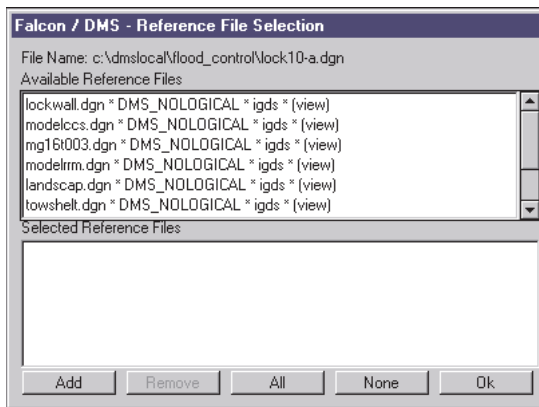Available Reference Files

```
lockwall.dgn * DMS_NOLOGICAL * igds * (view)
modelccs.dgn * DMS_NOLOGICAL * igds * (view)
mg16t003.dgn * DMS_NOLOGICAL * igds * (view)
modelrrm.dgn * DMS_NOLOGICAL * igds * (view)
landscap.dgn * DMS_NOLOGICAL * igds * (view)
towshelt.dgn * DMS_NOLOGICAL * igds * (view)
```

Selected Reference Files

Add	Remove	All	None	Ok

MDL—Basic Rules

The MicroStation (MDL) version of the Document Manager performs all the tasks involved in attaching reference files; the files are identified and the Attach Reference File command is launched from the Document Manager. However, reference files are detached with MicroStation's own Detach Reference command.

Because Falcon/DMS downloads all reference files as view_only files, there is no need to return the files to the server. Users are therefore responsible for deleting these local, view_only files when they are no longer needed—this, also, in no way affects the connection between the server-based original and its reference files.

The Reference File Refresh Command

When one designer checks in a file that another designer is using as a reference file, the server-based version of the file becomes more current than the version downloaded as a reference file. Falcon/DMS enables such reference files to be 'refreshed'; in other words, the local reference file can be updated with either the checked in version of the original, or a copy of the original that is itself 'refreshed' to the server. (The equivalent MicroStation tool is the Reload command.)

Reference Path Logicals, a Primer

Reference File Path Logical vs Reference File Logical Names

Take care not to confuse reference file path logicals with reference file logical names.

When an active file and its reference files are located in different directories, a reference file path logical is necessary to maintain the connections. Without a reference file path logical (or a specific value assigned to the MicroStation configuration variable MS_REFDIR), each time a designer opens a file whose reference files are located in a different directory, MicroStation will be unable to find and display the reference files.

How does this differ from a reference file logical name?

There is only one situation in which a reference file logical name is required—when a single file is attached as a reference file two or more times to the same active file. Because MicroStation must be able to distinguish between each attachment, it looks at the file's logical names to identify an attachment. (Because its twenty character length limit is more descriptive than the eight character limit of a filename, a reference file logical name can also be used optionally as a file description.)

The reference file path logical is central to the successful use of reference files. This MicroStation environment variable helps to maintain the connection between an active file and its reference files when the files are in different subdirectory locations. Falcon/DMS has four Reference Path Logicals options that impact its Attach Reference File command. These options are set from the Administrator's MicroStation Configuration dialog box:

❏ System Generated logicals
❏ Manager Defined logicals
❏ Keyin logicals
❏ None

Because MicroStation does not restrict the storage of files to preset locations, it relies on something called 'configuration variables' to find files that are maintained in directories of the user's choosing. MS_RFDIR is the variable MicroStation uses to locate reference files. MS_RFDIR is a member of the variable category called Primary Search Paths, and can point to as many directories as the user wishes. Search paths are not limited to the MS_RFDIR environment variable—search paths may also be defined as reference file *path logicals*. For example:

    n:\ventron\0017a\arch\

can be assigned a path logical of 'ven_a'. This path logical is stored in MicroStation's user or project configuration files. Because 'ven_a' now points to n:\ventron\0017\arch, a user can simply key in

    rf=ven_a:ref1.dgn

to attach n:\ventron\0017a\arch\ref1.dgn to an active file in a different directory, or on a different server, and MicroStation will be able to maintain that reference file attachment from one design session to the next.

Under certain circumstances the issue of reference file path logicals is just as important to Falcon/DMS as it is to MicroStation. If you use reference files and your installation meets any of the following criteria, you may need to implement some level of reference path logicals:

❏ You are not accessing files directly on the server—in other words, Falcon/DMS places a copy of the server-based original on your local workstation.
❏ Your database's primary index key is set to 'dirLevel1;Filename' or greater.
❏ Your active files and their reference files are not stored in the same directories.

Even if your organization meets all the criteria listed above, Falcon/DMS may be able to automatically handle reference file path logicals as a function of your database and its primary index key.

As you will see, to attach a design file to another as a reference file you must first select the file to be attached. Falcon/DMS next performs a database search. If a database record is not found for the selected file, you are unable to proceed. If a database record is found, Falcon/DMS may have enough directory path information to proceed without requiring a reference path logical. (This is because each design file's master directory and subdirectory information is stored in that document's database record.) For example:

❏ The Primary Index Key is 'dirLevel1;dirLevel2;Filename'
❏ You wish to attach ref1.dgn to active.dgn—the files are located in the following directories:
❏ Though ref1.dgn and active.dgn are in different 'dirLevel3' subdirectories, because 'dirLevel1' and 'dirLevel2' are the same for both (ventron and 0017-a), Falcon/DMS does not require a reference path logical. However, if you wish to attach ref2.dgn to active.dgn, a reference path logical is necessary because the 'dirLevel2' subdirectories are different (0018-c and 0017-a, respectively).

The 'Reference Path Logicals' Options

Falcon/DMS provides four Reference Path Logicals options. The options are set from the *MicroStation Configuration* dialog box and may be enabled in any combination.

The Manager Defined Logical has precedence over the other options. That is, should you enable multiple options that includes the Manager Defined Logical option, the remaining options are available only if a Manager Defined Logical does not exist for a selected path.

System Generated Logical:

❏ Falcon/DMS automatically generates a System Generated Logical when a reference file is attached. For example:

'n:\project1\mech\filename.dgn' is the file to be attached as a reference file.

❏ When the database's primary index key is dirLevel1;Filename, the reference file is attached with the following path logical specification:

DMS_PROJECT1:filename.dgn

❏ When the database's primary index key is dirLevel1;dirLevel2;Filename, the reference file is attached with the following path logical specification:

DMS_PROJECT1_MECH:filename.dgn

Manager Defined Logical:

A Manager Defined Logical is a user defined path logical from the Manager Defined Reference File Logicals dialog box. When a user attaches a reference file located in a subdirectory for which a path logical is defined, Falcon/DMS automatically assigns that logical to the reference file attachment. For example:

❏ VENTM ventron mech is the Manager Defined Logical for files located in the path (n):\ventron\mech.

❏ When a user attaches ref1.dgn, located in (n):\ventron\mech, as a reference file, the file is attach with the following specification: ventm:ref1.dgn.

To define a Manager Defined Logical:

1. In the Reference File Logicals region on the *MicroStation Configuration* dialog box, click the Configure button.

2. On the *Manager Defined Reference File Logicals* dialog box, click the Add button. In the Logical field, enter a logical name (all fields have an 8 character limit). In the 'master' and 'subdirectory' fields, define the path.

3. When finished, click the OK and Cancel Add buttons, then the Save and Exit buttons.

Keyin Logical:

A Keyin Logical allows the user to define the path logical at the time a reference file is attached. This option is provided so that the CADD manager can satisfy contractor or vendor reference path logical requirements. With Keyin Logicals the manager will not have to build and maintain 'vendor specific' Manager Defined Logical tables.

Because a Keyin Logical does not resolve reference file attachments when the database's Primary Index Key is 'dirLevel1;Filename' or greater, and if design files and their reference files are not stored in the same subdirectory, a System Generated or Manager Defined logical is required.

No Logical:

When No Logical is chosen, the database's Primary Index Key has a profound effect on the ability of Falcon/DMS to resolve reference file attachments.

When the index key is 'Filename', no reference path logical is required. Because each filename occurs only once in the corporate database, its database record contains all the path information necessary resolve reference file attachments.

This is not the case for a database with a Primary Index Key of 'DirLevel1;Filename' or greater. Any such database is unable to support reference file attachments that span 'Projects'. At the very least, design files must be located under the same master directory. For example:

❏ Primary Index Key: DirLevel1;Filename
❏ Active design file: n:\bentram\0017-a\arch\active.dgn
❏ Reference file: n:\bentram\0018-c\mech\ref. dgn

When the Primary Index Key is DirLevel1;Filename, Falcon/DMS insures that a filename will appear only once within each master directory path. Because active.dgn and ref.dgn have the same top level directory (DirLevel1 = bentram), the reference file attachment can be resolved.

Intelligent Data Link—IDL

IDL, or Intelligent Data Link, is a database feature designed to automatically define the contents of a design file's title block. Integrated with the Create Design File command, IDL uses information entered in the General Info, Directory & Keywords, and Title Block 'tabs' found on both the Create Design File dialog box and the Document Manager.

The Intelligent Data Link is based on an IDL template file that is automatically linked to a MicroStation seed file of the same name. Using MicroStation's text node technology, the IDL template file maps database fields to predefined text nodes in the linked seed file.

IDL is bidirectional — the text displayed on targeted text nodes in a MicroStation design file can be modified from the document's database record, or the database record can be modified by editing the text nodes with MicroStation's Edit Text command.

Configuring IDL is a two phase process

Phase 1: MicroStation

In a MicroStation design file, place the text nodes to which IDL will write database field information. This design file will be available to the user as a seed file from the Falcon/DMS Create Design File dialog box. The design file/seed file is given the same name as the IDL template created in Phase 2.

Phase 2: The Falcon/DMS Administrator

Step 1: Enable IDL from the MicroStation Configuration dialog box (in the File Formats/Applications environment)

Step 2: Define the IDL template file from the Intelligent Data Link Configuration (IDL) dialog box. This IDL template file is given the same name as the design file created in Phase 1.

Falcon/DMS is delivered with an IDL demonstration file set. In the <falcon_sdir>\ cfg\titleblk subdirectory you will find idl_demo.tbk, the IDL template file. In the <falcon_sdir>\seed subdirectory you will find idl_demo.dgn, the MicroStation design file that is used as the seed file for title block schemes defined in idl_demo.tbk.

To configure IDL

Phase 1

Configuring the MicroStation Seed File:

Pictured below are the 7 text nodes placed in the title block region of idl_demo.dgn; use this file as an example when creating your own idl seed files. For additional information on text nodes, consult the MicroStation documentation.
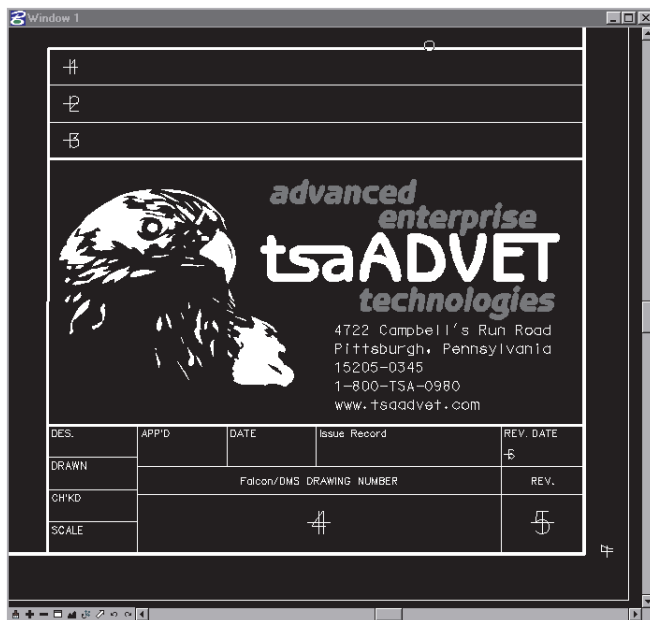

Phase 2

Step 1: Enabling IDL

1. Open the MicroStation Configuration dialog box. Choose Configure > File Formats/Applications > Edit Global Configuration on the Administrator. On the File Formats/Applications dialog box, double-click the 'igds' file format, then press Additional igds Parameters.

2. On the MicroStation Configuration dialog box, turn on Enable IDL for MDL. Press Save and Exit.

3. On the File Formats/Applications dialog box, press Cancel Edit and Exit.

You are now ready to define an IDL template file.

Step 2: Configuring an IDL template:

1. Choose Configure > Intelligent Data Link on the Administrator. The Intelligent Data Link Configuration (IDL) dialog box opens.
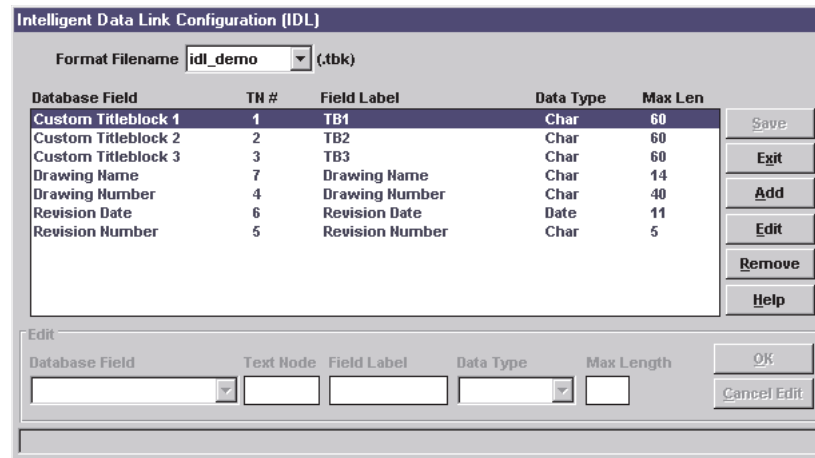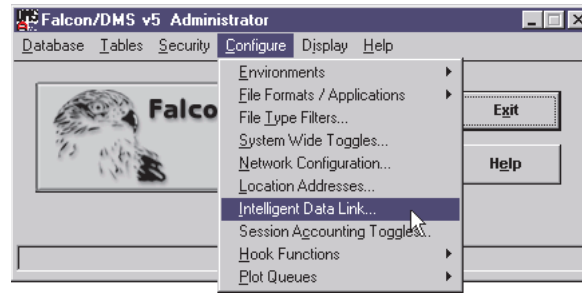
2. Select <New> from the Format Filename option menu on the Intelligent Data Link Configuration (IDL) dialog box.

A Falcon/DMS palette asks—"Do you want to clear the current IDL template?" Press Yes if you want a fresh template. Press No to write the current items to a new template where they can be added to, edited, or removed.

3. Key in a filename (there is an 8 character limit). Press OK.

4. In the Add region:

Select a field from the Database Field option menu.

In the Text Node field, key in the text node in the MicroStation seed file to which the database field maps.

If necessary, enter a Field Label—most fields are automatically assigned a label.
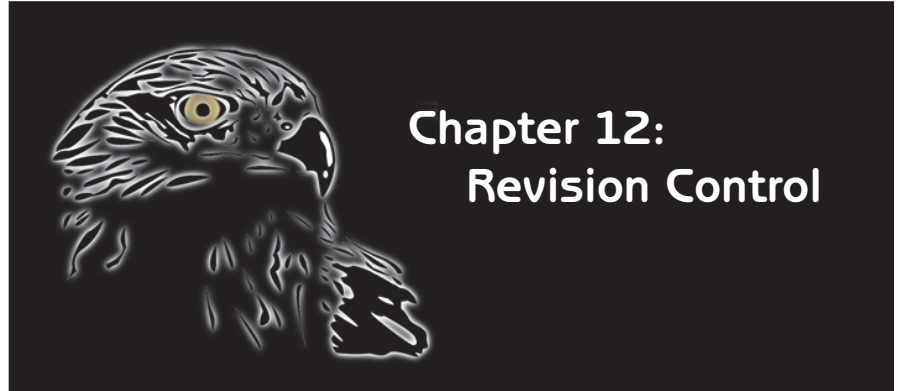
Select a data type (char, integer, or date) from the Data Type option menu.

Press OK. Repeat as needed. When finished, press Cancel Add.

5. Press Save and Exit.

Falcon/DMS

The Administrator

Chapter 12:
Revision Control

Revision Control is designed to track revisions in one of two modes—a History mode or an As-built/work-in-progress mode. The key difference between modes is in how each identifies the current revision, also referred to as the working document.

In the History mode, the current revision of a document is the original file and reflects the current state of work. It retains the original filename and has the highest number or letter assigned to its Rev# database attribute. When Revision Control generates a new revision in this mode, the current revision is copied and renamed using one of three filename format schemes explained on the following page. This copy is essentially a 'snapshot' of that moment's state of work and should be treated as a record document, not to be modified. To facilitate this, Revision Control can be configured to automatically assign a Document Status to each copy that sets an access mode of view_only or no_access.

In the As-built/work-in-progress mode, the current revision is a copy of the previous revision. The new working document is renamed using one of the three filename format schemes and the next available number or letter is assigned to its Rev# database attribute. Like History mode, the previous revision of an As-built document is a 'snapshot' of a state of work and should be treated as a record document, not to be modified.

Because Revision Control schemes are part of a File Format definition, each file format can have a unique Revision Control scheme and individual configurations can be enabled or not. A scheme can be configured to leave the previous revision in the current directory or move it to a subdirectory which Revision Control automatically creates. Finally, Revision Control can be configured to automatically create a new revision when a document is checked out or checked in, prompt for a revision, or allow users to create a revision at their discretion.

Revision Control

Revision Control is designed to track revisions in one of two modes—a History mode or an As–built/work–in–progress mode. The key difference between modes is in how each identifies the current revision.

History Mode

In the History mode, the current revision of a document is actually the original file—it reflects the current state of work and its Rev# database field will always have the highest number or letter. Each time Revision Control is used to generate a new revision of a document, Falcon/DMS creates a copy of the original and appends a notation to that copy's filename indicating, by number or letter, its revision level.

This copy is essentially a 'snapshot' of that moment's state of work and should be treated as a record document, not to be modified. To facilitate this, Revision Control can be configured to automatically assign a special Document Status to each new copy. To protect the revision 'history' from being modified, the Document Status is usually configured to set an access mode of view_only or no_access on each new copy. Additionally, if the 'move revision' option is enabled by the Falcon/DMS administrator, each new copy is automatically moved to a designated subdirectory, so that only the current revision remains in the project directory.

For example, lock10a.dgn is a document for which no revisions have yet been created, therefore the document's Rev# database field is blank. When the document's owner generates a new revision for lock10a.dgn, Falcon/DMS creates the copy using one of the following three filename format schemes:

❏ filename.ext.rev#   [e.g. lock10a.dgn.rev0 or lock10a.dgn.revA]
❏ filename.rev#.ext   [e.g. lock10a.rev0.dwg or lock10a.revA.dwg]
❏ filename.#   [e.g. lock10a.0 or lock10a.A]

The copy, let's say lock10a.dgn.rev0, is the 'snapshot' of that moment's state of work and is considered to be the previous revision. Because this is the first time Revision Control has been used for this document, the notation indicates that it is revision '0'. The administrator, in this case, has enabled the 'move revision' option, causing Falcon/DMS to move lock10a.dgn.rev0 to a subdirectory beneath the current directory.

Revision Control sets the Rev# for lock10a.dgn, to '1'. (Remember that in History mode the current revision retains the original filename and its Rev# will be the highest number or letter in the revision sequence.) Because its Rev# is currently '1', the next time Falcon/DMS generates a new revision for lock10a.dgn, the copy will be named lock10a.dgn.rev1, and the Rev# database attribute for lock10a.dgn will be changed to '2'.

As-Built/Work-in-Progress Mode

In the As–built/work–in–progress mode the current revision is the latest copy of a document—it reflects the current state of work and both it filename and its Rev# database field will always have the highest number or letter. (This is the opposite of the History mode, in which the current revision is the original document rather than the latest copy of the original.) Each time Revision Control is used to generate a new revision of a document, Falcon/DMS creates a copy of the previous revision and appends a notation to this copy's filename indicating, by number or letter, its revision level.

The previous revision is essentially a 'snapshot' of that moment's state of work and should be treated as a record document, not to be modified. The same Document Status options available for the History Mode are available for the As–built mode. When the 'move revision' option is enabled, each previous revision is automatically moved to a designated subdirectory, so that only the current revision remains in the project directory.

For example, lock11a.dgn is a the current revision. No revisions have yet been created, therefore its Rev# database field is blank. When the document's owner generates a new revision for lock11a.dgn in the As-built mode, Falcon/DMS creates a copy of this document using one of the following three filename format schemes:

❏ filename.ext.rev#   [e.g. lock11a.dgn.rev0 or lock11a.dgn.revA]
❏ filename.rev#.ext   [e.g. lock11a.rev0.dwg or lock11a.revA.dwg]
❏ filename.#   [e.g. lock11a.0 or lock11a.A]

The copy, let's say lock11a.dgn.revA becomes the current revision and its Rev# database field is set to 'A'. In this mode lock11a.dgn is the 'snapshot' of that moment's state of work and is considered to be the previous revision. The administrator, in this case, has also enabled the 'move revision' option, causing Falcon/DMS to move lock10a.dgn to a subdirectory beneath the current project directory.

The next time Falcon/DMS generates a new as-built revision for lock10a.dgn.revA, the new revision will be named lock10a.dgn.revB, its Rev# will be set to 'B', and lock10a.dgn.revA will be automatically moved to the subdirectory designated for previous revisions.

On the following pages we explain the configuration options for both Revision Control modes and list and illustrate the most common configurations by explaining what the users sees for two typical Revision Control definitions. The chapter concludes with an explanation of how Revision Control handles MicroStation reference files.

Configuring Revision Control

To open the Revision Control Configuration dialog box:

On Falcon/DMS Administrator, select Configure > File Fomats/Applications > Global Configuration. On the File Formats/Applications dialog box, double-click a file format. Press Revision Control Cfg.
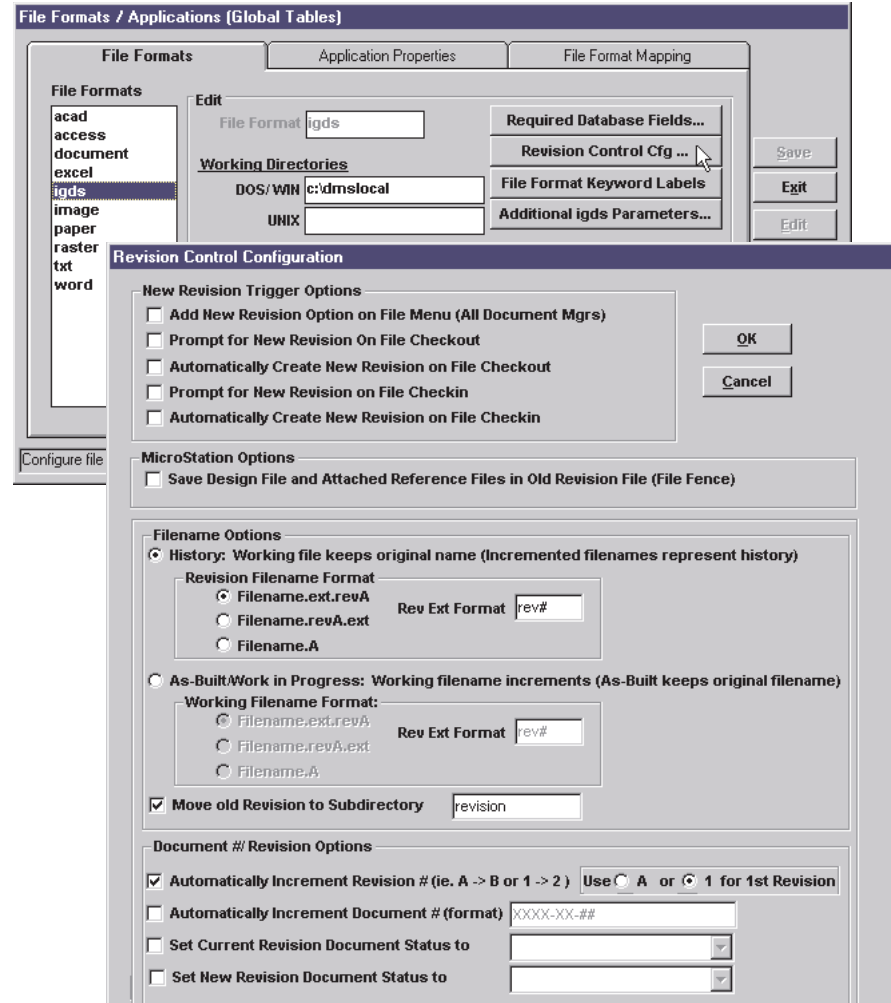
The Revision Control Configuration dialog box permits you to:

❑ Define the different options that trigger Revision Control.
❑ Choose a Revision Control mode—History or As–built/work–in–progress.
❑ Select a filename format and define the revision flag.
❑ Identify a directory to which previous revisions are moved.
❑ Define the Document # options.
❑ Set the appropriate Document Status Access for both current and previous revisions.

GROUP	OPTIONS
New Revision Trigger Options	Add New Revision Option on the File Menu (All Document Mngrs):
	Places the New Rivision command on the Document Manager's File menu. The New Revision command is available when a user selects a file of a file format for which Revision Control is configured. Enable this control when you wish users to make revisions at their discretion.
	Prompt for New Revision on File Checkout:
	When enabled, Falcon/DMS displays the New Revision dialog box when users checkout a file. Enable this control when you wish users to make revisions at their discretion..
	Automatically Create New Revision on File Checkout:
	When enabled, Falcon/DMS automatically creates a new revision when users check out a file.
	Prompt for New Revision on File Checkin:
	When enabled, Falcon/DMS displays the New Revision dialog box when users checkin a file. Enable this control when you wish users to make revisions at their discretion.
	Automatically Create New Revision on File Checkin:
	When enabled, Falcon/DMS automatically creates a new revision when users checkin a file.
MicroStation Options	Save Design File and Attached Reference Files in Old Revision File (File Fence):
	Allows user to save MicroStation design files and any attached reference file/files as a new revision; this operations uses the MicroStation File Fence command.
Filename Options	History: Working file keeps original name (incremented filenames represent history):
	In the History mode, the working document will always 'inherit' the original filename. Thus, the current revision is the document that reflects the last 'snapshot' of this working document. Any new revision will reflect the progress of work since the current 'snapshot', making the current revision one generation older.
	Revision Filename Format:
	Two filename formats are available. The first appends a user-defined new revision number or letter to the the filename and its current extension. The second replaces the filename extension with the new revision number or letter.
	As-Built/Work in Progress: Working filename increments (As-Built keeps original name):
	In the As-Built/Work in Progress mode, the working document's filename will always be an incremented version of the original filename. The original filename is 'owned' by the record copy of the document. In this mode the content of this record copy will be updated periodically to match that of the working document. In this scheme the current revision, the document with the highest incrementation, is the working document. Thus, any new revision is the next generation of the working document.
	Working Filename Format:
	Two filename formats are available. The first appends a user-defined new revision number or letter to the the filename and its current extension. The second replaces the filename extension with the new revision number or letter.
	Move Old Revision to Subdirectory:
	Enable this option to create a subdirectory into which Falcon/DMS places the new revision (History Mode) or the new working document (As-Built/Work in Progress mode). This option is probably more suitable for the History mode than the Work in Progress mode.
Document # Revision Options	Automatically increment Revision #:
	Automatically increments the Rev # value in the the document's database record.
	Automatically increment Document #:
	Automatically increments theDocument # value in the the document's database record.
	Set Current Revision Document Status To:
	Automatically sets a designated Document Status Access for the current revision.
	Set New Revision Document Status To:
	Automatically sets a designated Document Status Access for the new revision.

What the user sees

To illustrate the two Revision Control Modes and most common configuration for each, lets consider the procedures users follow to generate revisions. In the examples, the History mode will be configured such that the user generates a revision at his discretion, and the As–built mode will be configured to prompt the user to generate a revision when a document is checked out. The other 'prompt user' trigger option, 'Create new revision on file check in', is similar to the check out trigger option. The 'automatic' options are self explanatory.

History Mode

The following revision control workflow demonstrates the steps a user takes to create a new revision of a MicroStation design file when Revision Control is configured for History Mode and the trigger option is set to allow the user to generate a revision at his discretion. The following revision control settings are enabled:

❏ **Trigger Options:** Add New Revision Option on File Menu
❏ **Filename Options:** History
❏ **Filename Options > Revision Filename Format:** Filename.ext.revA
❏ **Filename Options > Revision Filename Format > Rev Ext Format:** rev#
❏ **Filename Options:** Move old Revision to Subdirectory—subdir name: revision
❏ **Document # Revision Options:**
  ❍ Automatically Increment Revision #; Use '1' for 1st Revision.
  ❍ Automatically Increment Document #: format—xxx-xx-##
  ❍ Set Current Revision Document Status Access to: revision
  ❍ Set New Revision Document Status Access to: active

From within MicroStation, the user would open a MicroStation design file, for example lock10a.dgn. Falcon/DMS checks out the selected document and opens it in Micro-Station. After editing the document, the user would determine when sufficient changes have been made to warrant creating a revision '1' of lock10a.dgn. *At such time the user must check in lock10a.dgn—In the current release, Revision Control can create revisions only for files with an Availability of 'On Server'.* To generate a revision of lock10a.dgn at his discretion, the user would perform the following procedure:

1. On the Document Manager, the user selects lock10a.dgn. Note that the Rev# database field for this document is blank, the Document # database field contains 'DACW37-98-A', and the Document Status is set to 'active'.
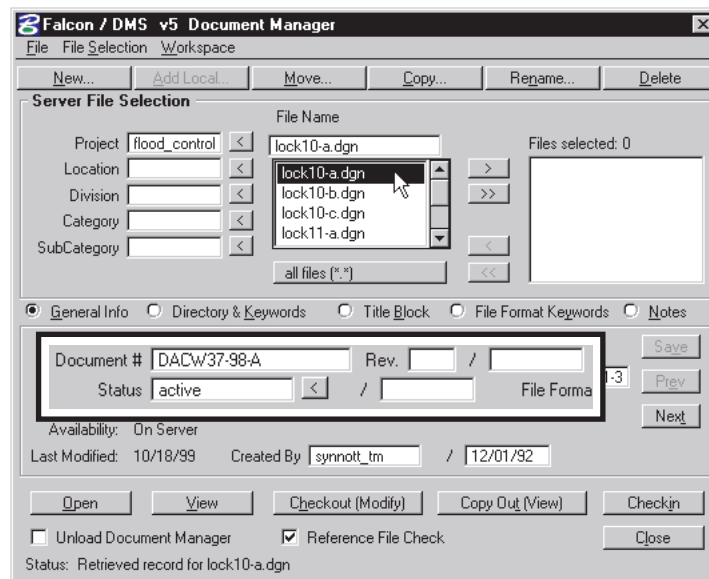
2. From the Document Manager's menu bar, the user selects File > New Revision. The New Revision form opens. On this form the user will confirm the parameters for the new revision and press OK.
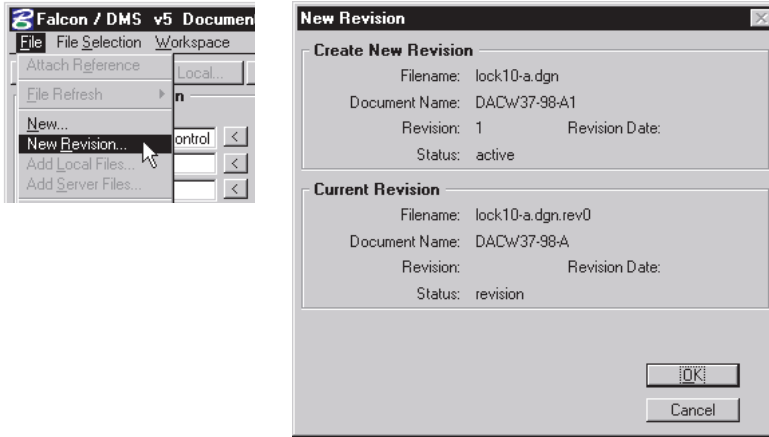
The result of this operation is shown below. Note that Rev# database field for lock10a.dgn contains '1', the Document # database field now reads 'DACW37-98-A1', and the Document Status remains as 'active'.
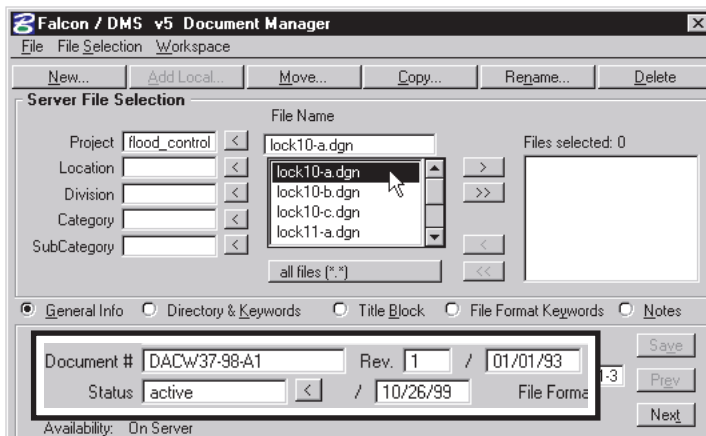
The previous revision would be moved to a 'revision' directory and renamed lock10a.dgn.rev0. Note that the Document Status is set to 'revision', a typical status that would most likely have an access mode of view_only or no_access.
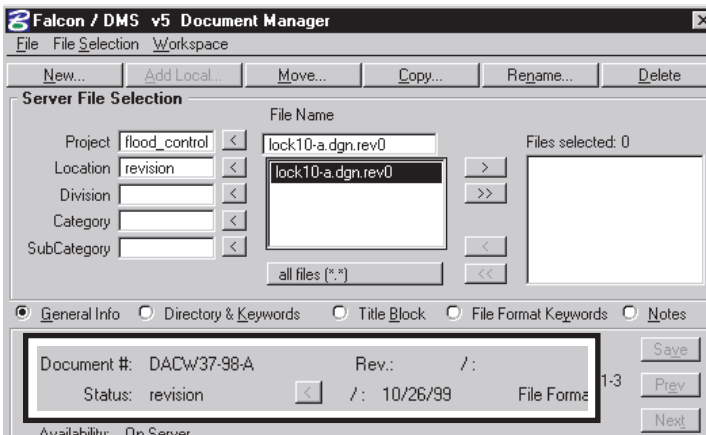
As-Built/Work-In-Progress Mode

The following revision control workflow demonstrates the steps a user takes to create a new revision of a MicroStation design file when Revision Control is configured for As–built/work–in–progress Mode and the trigger option is set to 'prompt on checkout'. The following revision control settings are enabled:

❏ **Trigger Options:** Prompt for New Revision On File Checkout
❏ **Filename Options:** As–built/work–in–progress
❏ **Filename Options > Revision Filename Format:** Filename.ext.revA
❏ **Filename Options > Revision Filename Format > Rev Ext Format:** rev#
❏ **Document # Revision Options:**
   ❍ Automatically Increment Revision #; Use 'A' for 1st Revision.
   ❍ Automatically Increment Document #: Format—xxx-xx-##
   ❍ Set Current Revision Document Status Access to: revision
   ❍ Set New Revision Document Status Access to: active

An important difference between this particular As–built mode configuration and the History mode configuration described on the preceding pages is that this As–built configuration does not move the previous revisions to a designated subdirectory. Rather, all previous revisions will remain with the current revision in the same project directory.

❏ **Filename Options:** Move old Revision to Subdirectory—NOT ENABLED

1. On the Document Manager, the user selects lock11a.dgn. Note that the Rev# database field for this document is blank, the Document # database field contains 'DACW37-98', and the Document Status is set to 'active'.
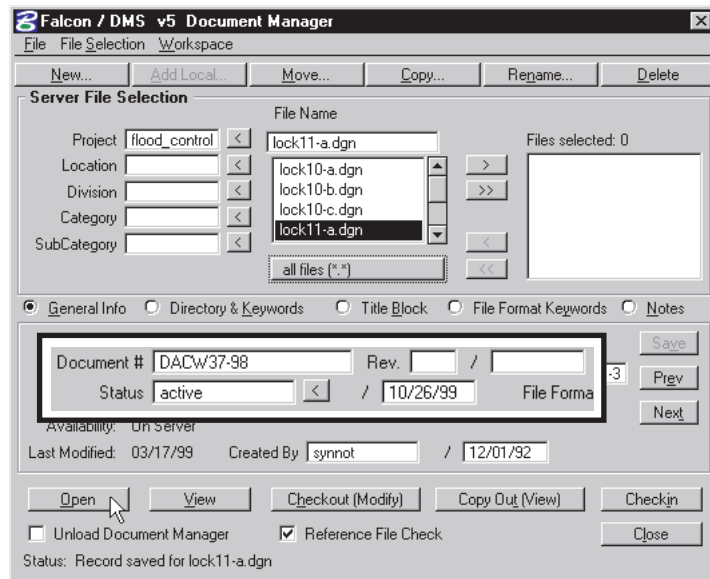
2. The user presses Open. The New Revision dialog box opens.

   Note: Because the Open command checks out the selected file as part of the open transaction, the current trigger option, 'prompt on checkout', recognizes this command as one that launches Revision Control and prompts the user.

On the New Revision form, the user will confirm the parameters for the new revision and select how the Open operation is to proceed:

❏ **Create & Checkout New Revision:** Creates and then opens or checks out the new revision

❏ **Checkout Current Revision:** Opens or checks out the current revision.

The result of this operation is shown below. Note that Rev# database field for lock11a.dgn contains 'A', the Document # database field now reads 'DACW37-98A', and the Document Status remains as 'active'.

All previous revisions will remain with the current revision in the same project directory. Note that the Document Status for lock11a.dgn is now set to 'revision', and would most likely have an access mode of view_only or no_access.

Revision Control and MicroStation Reference Files

When a revision is generated for a MicroStation design file to which reference files are attached, it is important that both the parent file and its reference files be accounted for in what is about to become a record document.

To insure that Revision Control properly generates complete record documents for MicroStation design files, you should enable the MicroStation Option on the Revision Control Configuration dialog box.
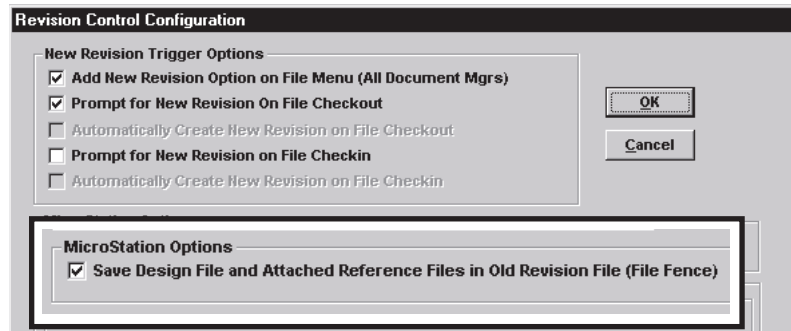
This option, 'Save Design File and Attached Reference Files', enables Revision Control the use MicroStation's File Fence capabilities to merge all reference files with the parent file as it generates the revision (record document). Be aware that only the graphic elements in reference files for which 'Display' is enabled, and located only on those levels that are visible, will be captured in this 'snapshot'.

Falcon/DMS

The Administrator

Chapter 13:
Reports
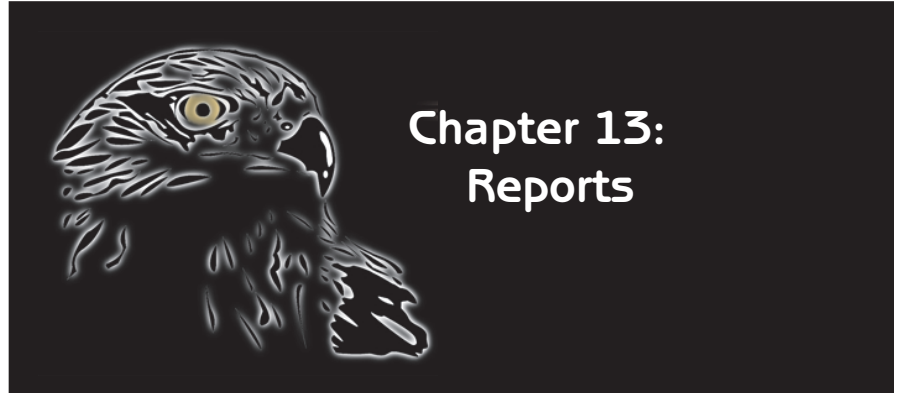
Falcon/DMS provides two report modules:

❏ **Main Database Reports:** The Main Database Reports module is part of the core product and is delivered with three pre defined reports.

❏ **Session Accounting Reports:** Session Accounting is an add-on report module that provides the Falcon/DMS administrator with the ability to track and report on a wide variety of document management activities and is delivered with two pre defined reports.

An especially powerful feature of the Session Accounting module is the ability to provide highly accurate, time-in-file accounting for tracking modification times in MicroStation design files. As part of this feature you can define a 'hibernation' parameter that suspends the time tracking after Session Accounting detects no cursor activity or keyboard input after a prescribed number of minutes. Session Accounting resumes the time-in-file tracking as soon as cursor activity or keyboard input is detected.

When Session Accounting is implemented it must be enabled for all workstations that use Falcon/DMS to access the data repository.

Both Main Database Reports and Session Accounting use a run-time version of Seagate's Crystal Reports as a report generator—to modify the delivered report formats or design your own will require a licensed version of Crystal Reports. However, because both the Main Database and the Session Accounting modules now allow you to define additional search criteria, the range of possible reports goes far beyond that available from the pre defined report formats.

Main Database Reports

The SQL version of Main Database Reports uses the database to store and generate report data. Falcon/DMS is delivered with three pre defined reports:

❏ **Drawing Status Report**, which generates a report listing all documents in the database and indicates the Drawing Status of each.

❏ **Drawings out to Another Location**, which generates a report of all documents that have been check out to a location with the Location Checkout command.

❏ **Files Checked out to Workstation**, when generates a report of all documents with an availability of 'On Workstation'. This report indicates the workstation to which the each document is checked out, the user to whom it is checked out, and the checkout and/or open date.

The SQL version of Main Database Reports allows use of the Falcon/DMS search engine to refine the criteria defined in each report. This is an option that can be toggled from the Options menu on the report form.

To generate a Main Database report:

1. In the Falcon/DMS Program Group, select Main Database Reports. The Falcon/DMS Reports application opens.
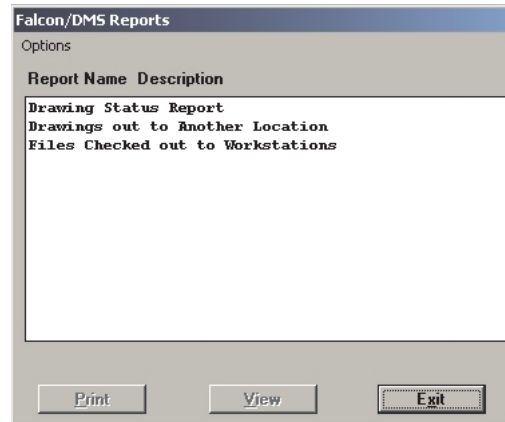
To allow use of the Falcon/DMS search engine to refine a report's criteria, select Options > Customize Search Criteria.

2. On the Reports form, select one of the three pre defined report formats.
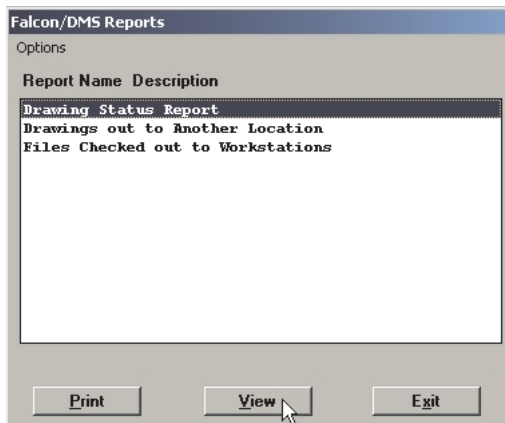
Press Print or View:

❏ Press Print to print directly to the default system printer.
❏ Press View to preview the report; you can then print the report from within the report preview.

When the Customize Search Criteria option is not enabled, the Print or View options generate an unfiltered report. When Customize Search Criteria is enabled, both Print and View display the Search Options dialog box.

2. From the Search Options dialog box, designate one of the following options:

❏ **Define Additional Search Criteria:** this option enables you to refine the reports search criteria. For example:

An unfiltered 'Drawing Status' report shows every document in the database, regardless of its Document Status; with the additional search criteria option you can narrow the result by specifying a particular status.

An unfiltered 'at Another Location' report shows all documents that have been sent off-site using the Location Checkout command; with the additional search criteria option you can narrow the result by specifying a particular location.

An unfiltered 'On Workstation' Status report shows every document that is checked out; with the additional search criteria option you can narrow the result by specifying a particular user or workstation.

Additionally, you can filter on directories and keywords as well as the date sensitive supplemental search criteria.

❏ **Use Report Search Criteria:** this option uses the report criteria established in the pre defined report and generally produces a very broad result.

Let's look an example of a report for each option.

The sample report pictured below is a Drawing Status report generated with the Use Report Search Criteria option. As you can see, the report page lists documents with different statuses.

The other option, Define Additional Search Criteria, allows you to use the Falcon/DMS search engine to refine addition criteria to narrow the results.

When the '..Search Criteria' option is selected, the Database Search dialog box opens. Use this form to refine the search.

For example, on the Database Search form specify a particular status, in this case 'construction'. Press Submit SQL.

The sample report pictured below lists only those documents with a Status of 'construction'.

Session Accounting Reports

Session Accounting is an add-on report module that provides the Falcon/DMS adminis-trator with the ability to track and report on a wide variety of document management activities and is delivered with two pre defined reports. The pre defined reports are:

❏ **File Activity**—generates a report of all activity for each document. The report breakdown shows activity by session type (modify, move, copy, etc.), user, workstation, start date/time, and cumulative session time per document.

❏ **Project Activity**—generates a report of all activity within a project. The report breakdown shows activity by user, start date/time, and cumulative session time per document.

Session Accounting Hibernation: An especially powerful feature of the Session Ac-counting module is the ability to provide highly accurate, time-in-file accounting of modification times in MicroStation design files. Session Accounting allows you to set a 'hibernation' parameters that suspend auditing when no cursor activity or keyboard input is detected after a prescribed number of minutes, and resumes the time-in-file tracking as soon as cursor activity or keyboard input occurs. The parameters are defined as environment variables that must be set on each workstation. The variables are:

dms_acct_hibernate = yes
dms_acct_hibernate_time = xx, where xx is the number of minutes of inactivity.

As with the Main Database Reports, Session Accounting allows use of the Falcon/DMS search engine to refine the criteria defined in each report. This is an option that can be toggled from the Options menu on the report form.

Session Accounting is enabled and configured from the Administrator's Session Account-ing Definition dialog box.

To enable and configure Session Accounting:

From the Administrator, choose Configure > Session Accounting Toggles.

The Session Accounting Definition dialog box contains two tabs:

❑ The Table Definition tab is used to enable Session Accounting and to indicate which session times are to be captured and which operations are to be tracked.

❑ The Database Fields tab indicates the database fields that are available on the Session Accounting version of the Database Search form. (This form is available only when Session Accounting data is written to and retrieved from the database.) When Session Accounting is configured to write directly to an ASCII file instead of the database, the Database Fields tab is used to define which attribute data is written to the ASCII file.

The Table Definition Tab

TABLE DEFINITION TAB ['Use RDBMS...' enabled]	
Category	**Description**
Use RDBMS for Session Accounting	When enabled, this control causes all Session Accounting information to be written to the relational database (the options listed in the Table Format and Location regions, and all options on the Database Fields tab, are locked when this control is enabled).
Enable Session Accounting	When enabled, Session Accounting is active and tracking the functions specified in the Functions Captured region on this tab.
Table Format	The Table Format controls are available only when the 'Use RDBMS for Session Accounting' option is disabled.
Table Location	The Table Location controls are available only the 'Use RDBMS for Session Accounting' option is disabled.
Functions Captured	The functions that Session Accounting tracks are divided into three categories: Session Times: Cumulative time is tracked for the following: Login: Available when Security > Global Security Settings > Require Login is enabled. Modifying Documents: Tracks time-in-file when a user opens a document through Falcon/DMS. For MicroStation sessions, the time-in-file tracking can be suspended by defining a 'hibernation' parameter. See 'Session Accounting Hibernation'. Viewing Documents: Tracks time-in-file when a user views a document through Falcon/DMS. User Functions: Session Accounting identifies the document, operation, user, and date/time for the specified Document Manager functions. The functions listed are considered to be standard user level operations. Manager Functions: Session Accounting identifies the document, operation, user, and date/time for the specified Document Manager functions. The functions listed are considered to be administrator level operations.

When the 'Use RDBMS for Session Accounting' check box is not enabled, Session Accounting does not write the accounting data to the database. Rather, it writes all auditing information to an ASCII file formatted according to the Table Format and Table Location parameters. The file, sessacc1.dat is placed in the <falcon_sdir>/acctdata server directory, or in the local <falcon_dir/acctdata directory, or both, depending upon the options set in the Table Location region. Typically, this file is then used in a spreadsheet application. Please note that Session Accounting will not simultaneously write to the database and to this ASCII file.

TABLE DEFINITION TAB ['Use RDBMS...' not enabled]	
Category	**Description**
Use RDBMS for Session Accounting	When not enabled, this control causes all Session Accounting information to be written to a data file that is formatted based on the controls located in the Table Format and Location regions. Additionally, the options on the Database Fields tab are unlocked. .
Table Format	The Table Format controls are available only when you disable the 'Use RDBMS for Session Accounting' option. The control allows you to specify either a fixed length or a text delimited data file, and to specify the separator or delimiter.
Table Location	The Table Location controls are available only when you disable the 'Use RDBMS for Session Accounting' option. Session Accounting can write the data file, sessacc1.dat, to the application server or to the Falcon/DMS administrator's local hard disk.

The Database Fields Tab

When 'Use RDBMS for Session Accounting' check box is enabled, the Database Fields tab is locked and cannot be modified. The enabled options on this form correspond to the fields available on the Session Accounting version of the Database Search form, pictured below.

When 'Use RDBMS for Session Accounting' check box is not enabled, the Database Fields tab can be modified to allow for additional database information to be written to
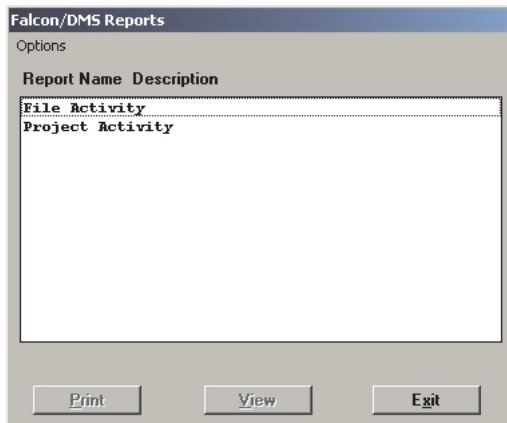
sessacc1.dat, the ASCII file.

When 'Use RDBMS for Session Accounting' check box is not enabled you will not use the pre defined Session Accounting reports, as these reports rely on the database to generate the results. Instead, you will typically use a spreadsheet application to generate reports and/or massage the Session Accounting data written to sessacc1.dat.

DMS

Session Accounting Reports

To generate a Session Accounting report [when 'Use RDBMS for Session Accounting' is enabled]:

1. In the Falcon/DMS Program Group, select Session Accounting Reports. The Falcon/DMS Reports application opens.

Falcon/DMS Reports

Options

Report Name Description

File Activity
Project Activity

Print   View   Exit

To allow use of the Falcon/DMS search engine to refine a report's criteria, select Options > Customize Search Criteria.

Falcon/DMS Reports

Options

Customize Search Criteria
Display Report Names

Project Activity

Print   View   Exit

2. On the Reports form, select one of the three pre defined report formats.

Press Print or View:

❏ Press Print to print directly to the default system printer.
❏ Press View to preview the report; you can then print the report from within the report preview.

When the Customize Search Criteria option is not enabled, the Print or View options generate an unfiltered report. When Customize Search Criteria is enabled, both Print and View display the Search Options dialog box.

2. From the Search Options dialog box, designate one of the following options:

❏ **Define Additional Search Criteria:** this option enables you to refine the reports search criteria. For example:

An unfiltered 'File Activity' report lists every document in the database; with the additional search criteria option you can narrow the result by specifying a particular Session Type (i.e.., modify, add server, copy, rename, etc.) for a specific file.

An unfiltered 'Project Activity' report organizes all documents by project and shows activity by user, start date/time, and cumulative session time per document; with the additional search criteria option you can narrow the result by specifying a subset of projects.

Additionally, you can filter on directories and keywords. However, unlike Main Database Reports, the Session Accounting version of the Database Search form does not support the criteria normally found on the Supplemental tab.

❏ **Use Report Search Criteria:** this option uses the report criteria established in the pre defined report and generally produces a very broad result.

Let's look an example of a report for each option.

The sample report pictured below is a File Activity report generated with the Use Report Search Criteria option. As you can see, the report page lists multiple files and numerous sessions.

File Activity

Filename	Session Type	Dir Level1	Dir Level2	User	Workstation	Start Date/Time	Session Time (
904113wk.dgn							
	Add-Rec	904113		thomas	SNOWMASS	1999/11/03 09:09:50.00	
	Moved-To	ntdot		thomas	SNOWMASS	1999/11/03 09:10:01.00	
	Moved-From	904113		thomas	SNOWMASS	1999/11/03 09:10:01.00	
	Add-Rec	dgn		thomas	SNOWMASS	1999/11/03 12:04:01.00	
	Moved-To	dgn		thomas	SNOWMASS	1999/11/03 12:03:21.00	
	Moved-From	ntdot		thomas	SNOWMASS	1999/11/03 12:03:21.00	
	Checkout	dgn		thomas	SNOWMASS	1999/11/03 12:14:04.00	
	Modify	dgn		thomas	SNOWMASS	1999/11/03 12:14:07.00	0.02
	Modify	dgn		thomas	SNOWMASS	1999/11/03 12:18:47.00	0.03
	Checkin	dgn		thomas	SNOWMASS	1999/11/03 12:21:42.00	
	Moved-To	904113		thomas	SNOWMASS	1999/11/03 12:21:49.00	
	Moved-From	dgn		thomas	SNOWMASS	1999/11/03 12:21:49.00	
	Checkout	904113		thomas	SNOWMASS	1999/11/03 12:31:30.00	
	Modify	904113		thomas	SNOWMASS	1999/11/03 12:31:32.00	0.01
	Checkin	904113		thomas	SNOWMASS	1999/11/03 12:32:19.00	
	Moved-To	dgn		thomas	SNOWMASS	1999/11/04 08:38:18.00	
	Moved-From	904113		thomas	SNOWMASS	1999/11/04 08:38:18.00	
	Moved-To	904113		thomas	SNOWMASS	1999/11/04 08:40:09.00	
	Moved-From	dgn		thomas	SNOWMASS	1999/11/04 08:40:09.00	
				Total Activity Time for : 904113wk.dgn			0.06
904113xl.alg							
	Add-Rec	904113		thomas	SNOWMASS	1999/11/03 09:12:33.00	
	Moved-To	ntdot		thomas	SNOWMASS	1999/11/03 09:12:54.00	
	Moved-From	904113		thomas	SNOWMASS	1999/11/03 09:12:54.00	
	Moved-To	904113		thomas	SNOWMASS	1999/11/04 14:44:06.00	
	Moved-From	ntdot		thomas	SNOWMASS	1999/11/04 14:44:06.00	
				Total Activity Time for : 904113xl.alg			0.00
904113z.2dp							
	Add-Rec	904113		thomas	SNOWMASS	1999/11/03 09:21:24.00	
	Moved-To	ntdot		thomas	SNOWMASS	1999/11/03 09:21:35.00	

| ◄◄ | ◄ | 44 of 71 | ► | ►► | Cancel | Close | 1657 of 1657 | Total:1657 | 100% |

The other option, Define Additional Search Criteria, allows you to use the Falcon/DMS search engine to refine addition criteria to narrow the results.

Search Options
- ⦿ Define Additional Search Criteria
- ○ Use Report Search Criteria

OK   Cancel
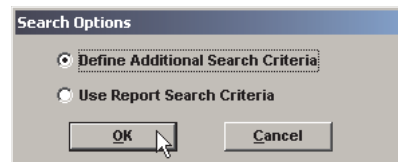
When the '..Search Criteria' option is selected, the Session Accounting version of the Database Search dialog box opens. Use this form to refine the search.