

TURNOVER® FOR ISERIES

Managing IFS Objects

Publication number

TIAM-1010-00 (October 2018)

Information in this publication is subject to change. Changes will be published in new editions or technical newsletters.

Documentation set

The documentation relating to this product includes:

- Getting Started with TURNOVER® for iSeries Client
- Getting Started with TURNOVER® for iSeries
- TURNOVER® for iSeries User Guide
- TURNOVER® for iSeries Developer's Guide
- Application Planning Guide
- Managing IFS Objects with TURNOVER® for iSeries
- TURNOVER® for iSeries PWM Quick Reference
- Synchronizer User Guide

Copyright notice

TURNOVER® for iSeries (the Programs and associated materials) is a proprietary product of UNICOM Systems, Inc. The Programs have been provided pursuant to License Agreement containing restrictions on their use. The programs and associated materials contain valuable trade secrets and proprietary information of UNICOM Systems, Inc. and are protected by United States Federal and non-United States copyright laws. The Programs and associated materials may not be reproduced, copied, changed, stored, disclosed to third parties, and distributed in any form or media (including but not limited to copies on magnetic media) without the express prior written permission of UNICOM Systems, Inc., 15535 San Fernando Mission Blvd., Suite 310, Mission Hills, CA 91345 U.S.A.

TURNOVER® for iSeries

Copyright © 2005-2018 UNICOM Systems, Inc. All rights reserved. No part of this Program may be reproduced in any form or by electronic means, including the use of information storage and retrieval systems, without the express prior written consent and authorization of UNICOM Systems, Inc.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, without prior written permission from UNICOM Systems, Inc.

Trademark acknowledgements

The following are trademarks or registered trademarks of IBM Corporation: APPC, APPN, iSeries, System i, IBM i, OfficeVision, and SNA.

Product and company names mentioned herein may be the trademarks or registered trademarks of their respective owners.

Table of Contents

CHAPTER 1: PREFACE	V
<i>Who Should Use this Guide</i>	<i>v</i>
<i>UNICOM Systems, Inc. Recommendations</i>	<i>v</i>
<i>UNICOM Systems, Inc. Cautions</i>	<i>v</i>
<i>The TURNOVER® for iSeries Information Library</i>	<i>vi</i>
CHAPTER 2: OVERVIEW	1
SUMMARY OF TURNOVER® FOR ISERIES CHANGES FOR IFS.....	1
<i>Global defaults</i>	<i>1</i>
<i>Global type codes</i>	<i>1</i>
<i>Application definitions</i>	<i>2</i>
<i>Commands and APIs</i>	<i>2</i>
<i>IFS object information.....</i>	<i>2</i>
<i>IFS exit points.....</i>	<i>2</i>
<i>Help text</i>	<i>2</i>
IFS OBJECT AUTHORITY	3
CHAPTER 3: SETTING UP TURNOVER® FOR ISERIES TO MANAGE IFS OBJECTS.....	5
GLOBAL DEFAULTS FOR IFS	5
<i>IFS Adopted authority user</i>	<i>5</i>
<i>IFS Archive subdirectory.....</i>	<i>6</i>
GLOBAL TYPE CODES FOR IFS.....	6
APPLICATION DEFAULTS AND RULES.....	8
<i>IFS Path defaults.....</i>	<i>9</i>
<i>IFS Line defaults</i>	<i>9</i>
<i>IFS Application type codes.....</i>	<i>13</i>
SUMMARY	13
CHAPTER 4: HOW TO MANAGE IFS OBJECTS	15
SUMMARY OF COMMANDS FOR MANAGING IFS OBJECTS	15
CHECKING OUT IFS OBJECTS (TCHKOUTIFS).....	16
<i>Checking In IFS Objects (TCHKINIFS).....</i>	<i>17</i>
CHANGING IFS OBJECTS	18
PROMOTING IFS OBJECTS.....	18
<i>Summary of IFS functions in Form Maintenance.....</i>	<i>18</i>
<i>Creating a form</i>	<i>18</i>
<i>Adding IFS objects to your form (TADDIFSLIN and TADDIFSLNI)</i>	<i>19</i>
<i>Linking pre-run and post-run commands to IFS form lines</i>	<i>20</i>
<i>About writing programs for IFS-related form processing.....</i>	<i>21</i>
RUNNING YOUR IFS FORM.....	22
SUMMARY	22
CHAPTER 5: DISTRIBUTING IFS OBJECTS.....	23
DISTRIBUTION-RELATED SUBDIRECTORIES FOR IFS OBJECTS	23
OTHER PREPARATIONS FOR DISTRIBUTING IFS OBJECTS.....	24
CHAPTER 6: FINDING TURNOVER® FOR ISERIES INFORMATION ABOUT IFS OBJECTS.....	25
IFS OBJECT HISTORY	25
<i>Archive history</i>	<i>25</i>
<i>Checkout history.....</i>	<i>27</i>
<i>Distribution history</i>	<i>27</i>

TURNOVER® for iSeries Managing IFS Objects with TURNOVER® for iSeries

Table of Contents

CHECKED OUT IFS OBJECTS.....	28
SUMMARY	29
IFS EXIT POINTS.....	32
IFS 2: POST CHECKOUT EXIT	32
<i>Example</i>	32
IFS 3: MANUAL CHECK-IN EXIT	32
<i>Example 1</i>	33
<i>Example 2</i>	33
IFS 4: ADD LINE EXIT	33
<i>Example</i>	33
IFS 5: PRE-CHECKOUT EXIT	33
<i>Example</i>	34
IFS 7: POST ADD LINE EXIT.....	34
<i>Example</i>	34
OVERVIEW	37
CONFIGURING DEFAULT LINKED COMMANDS.....	37
<i>Step 1: Check your global CLASS (or JAR or ZIP) type code</i>	38
<i>Step 2: Turn on automatic linking at the application level</i>	39
VERIFY YOUR SETUP	39
OVERVIEW	41
SYSTEM REQUIREMENTS	41
PREREQUISITES	41
CONFIGURING FOR WINDOWS FILE MANAGEMENT.....	42
TROUBLESHOOTING WINDOWS SUPPORT	42
PREREQUESITES	45
SUPPORTED OPERATING SYSTEMS.....	45
BACKGROUND	45
CONFIGURING TURNOVER® FOR iSERIES SUPPORT FOR UNIX.....	46
<i>Step 1: Define Exports on the UNIX System(s)</i>	46
<i>Step 2: Verify NFS service and firewall rules</i>	47
<i>Step 3: Mount Directories on IBM i systems</i>	47
<i>Step 4: Configure TURNOVER® for iSeries global defaults</i>	49
<i>Step 5: Create a TURNOVER® for iSeries authority list for use by NFS systems</i>	50
SUMMARY	51
ADDITIONAL RESOURCES	52
INDEX	1

CHAPTER 1: PREFACE

This guide provides instructions for installing TURNOVER® for iSeries on your IBM i system (for the first time), upgrading to TURNOVER® for iSeries, and applying changes to an existing TURNOVER® for iSeries release.

Who Should Use this Guide

This guide is intended for the TURNOVER® for iSeries Administrator and the IBM i System Administrator.

You, the reader of this guide, should have a general knowledge of the IBM i, be familiar with Security Administration, and understand the organization of your company's applications and databases. You should also have some understanding and experience with Command Language (CL).

UNICOM Systems, Inc. Recommendations

Throughout this guide you will be given a complete explanation of TURNOVER® for iSeries features and procedures to help you install the TURNOVER® for iSeries system quickly. Where appropriate, we have provided additional notes that contain our recommendations to help you with the decisions you need to make. We clearly label our recommendations like this:

UNICOM Systems, Inc. Recommends

Our first recommendation is probably the most important. We suggest that you assign someone to be your TURNOVER® for iSeries Administrator. This person does not need to be a programmer – in fact, most data processing auditors recommend that you assign someone outside of programming to act as your TURNOVER® for iSeries Administrator. This person might be an operations manager, quality assurance specialist, or database administrator. Ideally, this person understands the technical issues related to system security, data integrity, and object management, and has some programming experience.

UNICOM Systems, Inc. Cautions

In every system there are choices – some choices may come back to haunt you later, especially where your production environment is concerned. We have provided cautions whenever we felt that a particularly important choice needs to be made. Cautions look like this:

UNICOM Systems, Inc. Caution!

Although we have provided you with the capability to move objects (MO) and create duplicate objects (CD) from test libraries to production libraries, we recommend that you use the option CSCO (Copy Source to production and Compile Object) to ensure that the object is always created from the source member in your production library and that the correct files are used when creating the object.

PREFACE

The TURNOVER® for iSeries Information Library

Other sources of information about TURNOVER® for iSeries include:

- The TURNOVER® for iSeries information library, consisting of this *Getting Started with TURNOVER® for iSeries* manual, a user guide, and a tutorial:
 - o *Getting Started with TURNOVER® for iSeries* provides instructions for installing TURNOVER® for iSeries on your computer (for the first time), upgrading to a new release, and applying changes to an existing release.
 - o The *TURNOVER® for iSeries Tutorial* presents an example of a TURNOVER® for iSeries application, and walks you through the steps you would typically perform in your actual working environments.
 - o The *TURNOVER® for iSeries Developer's Guide* contains topics of specific interest to software developers. It also serves as an effective introduction to TURNOVER® for iSeries.
 - o The *Synchronizer User Guide* describes all of the features of the Synchronizer system, including the source compare and merge programs.
- The *TURNOVER® for iSeries Application Planning Guide* provides information to help you understand TURNOVER® for iSeries and TURNOVER® for iSeries application definitions. (This information is user interface independent.)
- The TURNOVER® for iSeries Supplements and Technical Bulletins are articles on specialized or advanced topics.
- The TURNOVER® for iSeries Interface Guides describe how to use TURNOVER® for iSeries to manage objects created by other products.¹
- The *Getting Started with the TURNOVER® for iSeries Client* guide provides installation and configuration instructions for using the Eclipse-based TURNOVER® for iSeries plug-ins, either in the standalone TURNOVER® for iSeries Client or within an Eclipse-based IDE such as IBM's RDi.
- *TURNOVER® for iSeries Supplement #66: Implementing TURNOVER® for SVN* provides complete information for installing and configuring the TURNOVER® for SVN version control product for your Web/PC/Java development projects. This product integrates seamlessly with the Eclipse-based TURNOVER® for iSeries plug-ins running either in the TURNOVER® for iSeries Client or another Eclipse-based IDE such as IBM's RDi.

In addition, Help text (both field sensitive and extended help) is available for every display. Position the cursor to any field, or to the panel title, and press the F1/Help key.

¹ For a complete list of products with which TURNOVER® for iSeries interfaces, visit the [SoftLanding website](#) or contact your UNICOM Systems, Inc. Sales Representative.

All TURNOVER® for iSeries documentation is available as Adobe™ Acrobat® PDF files within the TurnOverDocsClients.zip file. Using the Acrobat® Reader with Search (also provided), you can search, view, and print TURNOVER® for iSeries information.

CHAPTER 2: OVERVIEW

IFS stands for *integrated file system*. A *file system* is the way a computer's operating system stores data in logical units called files, directories, libraries, and objects. The IBM i operating system supports several different file systems, and includes a *common integrated file system interface* for accessing data stored in any of the supported file systems.

TURNOVER® for iSeries lets you manage objects stored in any of the IBM i-supported file systems² including IBM i V5R4 and beyond. This capability extends TURNOVER® for iSeries' change management expertise into the next century and beyond by opening the doors to both native and non-native IBM i technologies such as:

- JAVA, HTML, and other Internet-related applications;
- PC-based word or graphic processing applications;
- Client/Server development tools.

SUMMARY OF TURNOVER® FOR ISERIES CHANGES FOR IFS

TURNOVER® for iSeries' IFS support is initially API-driven. (Support for the Programmer Worklist will be added later.) As you might expect, it requires you to adjust the information³ in certain vital TURNOVER® for iSeries components.

Global defaults

There are new Global Default values for IFS on the TURNOVER® for iSeries Global Defaults panel.

- **IFS Adopted Authority User.** This default value controls what user profile's authority TURNOVER® for iSeries adopts during form processing so it can create objects in new locations and assign authorities to them there.
- **IFS Archive subdirectory.** This default value lets you set a default subdirectory for archiving IFS objects. You can override this value in the application definition.

For more information about these defaults, refer to page 5.

Global type codes

Sample IFS type codes are shipped with TURNOVER® for iSeries for IFS support. For detailed information, refer to page 6.

² For a description of the file systems supported by the operating system version your computers are running, read the **IBM i Integrated File System Introduction** publication provided with your operating system documentation.

³ Wherever possible, function key F8 has been used in the panel interface for toggling to the screens where you supply information for IFS objects. (In one case, an existing **F8** key assignment was reassigned to another key.)

CHAPTER 2: OVERVIEW

Application definitions

- Instead of Library defaults, IFS objects require ***Path defaults***. You specify these using **F8** on the ***Library Defaults*** panel.
- Form line requirements are different for IFS objects. You can specify form line default information for IFS objects using **F8** on the ***Line Defaults*** panel.
- IFS type codes must be included in your application's type code list.

For detailed information about the Application Definition requirements for IFS, refer to the descriptions beginning on page 8.

Commands and APIs

New commands exist for checking IFS objects in and out, for adding them to forms, and for linking commands to IFS form lines. Detailed information about these commands begins on page 15.

IFS object information

The ***Object History*** panels now contain a function key toggle for IFS object information, including checkout history, archive history and management, and distribution history. Detailed information begins on page 25.

The ***Check out/Check in source*** menu also includes a new option for viewing checked out IFS objects. Read about it on page 28.

IFS exit points

Exit points for custom processing at various stages of IFS object management are available when you press F8 on the ***Work with Exits*** panel. They include pre- and post-checkout exits, pre- and post-line exits, and a check-in exit. Detailed information starts on page 32.

Help text

Complete online help text is available for all new and changed panels.

IFS OBJECT AUTHORITY

During a TURNOVER® for iSeries form run, TURNOVER® for iSeries adopts ownership of objects in order to create them in new places and set authority parameters for them in the new location. This concept does not apply to IFS commands. Instead, you specify in your TURNOVER® for iSeries Global Defaults a specially-created user profile that TURNOVER® for iSeries uses to run IFS commands (parameter ***IFS Adopted authority user***). (Note that this user profile has special definition requirements; be sure to read about them on page 5.)

When creating IBM i file system objects, TURNOVER® for iSeries can use existing objects known as *reference objects* to set authority parameters. This is because the GRTOBJAUT command's REFOBJ parameter lets you specify an object to use as an authority model. IFS object authorities are assigned using a different command, called CHGAUT. It has no parameter for specifying reference objects.

Instead, TURNOVER® for iSeries applies IFS object authorities using a user profile-based authority system. IBM i system entities known as *authorization lists* and *primary groups* control who has access to what file system directories. (Authorization lists and primary groups are two different ways of grouping user profiles for the purpose of assigning authority to objects.)

For these reasons, the IFS line defaults have a parameter for a new TURNOVER® for iSeries entity called a *TURNOVER® for iSeries authority list* (see Figure 1). The TURNOVER® for iSeries IFS authority list is a named collection of data and object authority values that are assigned using primary groups, authorization lists, and the *PUBLIC system authority designation. When combined into an authority list, these values control assignment of IFS object authority at each level of your application. You can create a different TURNOVER® for iSeries authority list for each level of your application, if necessary.

CHAPTER 2: OVERVIEW

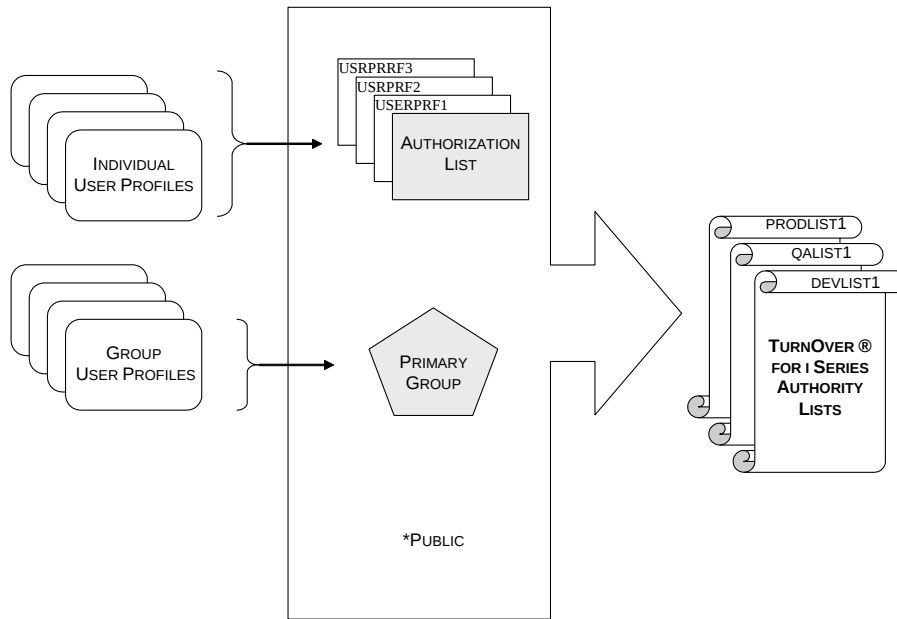


Figure 1: TURNOVER® for iSeries Authority Lists

CHAPTER 3: SETTING UP TURNOVER® FOR iSERIES TO MANAGE IFS OBJECTS

Before you begin using TURNOVER® for iSeries to manage IFS objects, you must configure some TURNOVER® for iSeries Global Default values, some TURNOVER® for iSeries global type codes, and some parts of any application definition you plan to use for managing IFS objects. This chapter contains detailed instructions and advice for adjusting these values.

GLOBAL DEFAULTS FOR IFS

Two global TURNOVER® for iSeries defaults apply to IFS:

- *IFS Adopted authority user*
- *IFS archive subdirectory.*

These are the last two defaults on the SETGLOBE panel⁴:

Set Global TURNOVER Defaults (SETGLOBE)

Type choices, press Enter.

Manage S/36 object types	> N	*SAME, Y, N
Allow group checkout	> N	*SAME, Y, N
Use Delete Test Object rule . . .	> Y	*SAME, Y, N
Company name	> 'Your Company'	
Dft pre/post run cmd lib list . .	> *CURRENT	*SAME, *CURRENT, *APPL
Specify method of distribution . .	> *DISTRIB	*SAME, *FORM, *DISTRIB
IFS Adopted authority user . . .	>	Name, *SAME, *CURRENT
IFS Archive subdirectory	> DEFAULT	Character value

Bottom

F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

IMPORTANT!!

This user profile has specific authority requirements. See description below.

IFS Adopted authority user

Use this parameter to control whether or not TURNOVER® for iSeries adopts the authority of a specially configured user profile on your system during checkout processing and promotion runs, as follows:

⁴ Main menu option 8 (Utility Menu), then option 5.

<u>Value</u>	<u>Description</u>
Name	If you specify a profile, TURNOVER® for iSeries will adopt that profile's authority when moving or creating objects and assigning object authorities. If you plan to have TURNOVER® for iSeries adopt authority, you should create a user profile specifically for the purpose. The profile <u>must</u> have *ALLOBJ authority or be a member of *SECOFR user class, or have appropriate authority for the IFS objects that TURNOVER® for iSeries will manage.
*CURRENT	TURNOVER® for iSeries uses the authority settings of the user associated with the job to move or create objects and assign authorities. The user must be able to manage object existence and authority, or processing will fail.

IFS Archive subdirectory

Specify a default subdirectory name for all archived IFS objects. You can override this value at the application level; look for this same parameter on the *Application Defaults and Rules* panel.

All IFS objects are archived in subdirectories of the /TURNOVER/ARCHIVE subdirectory, in the form:

/TURNOVER/ARCHIVE/ssssssss/ffffff

where ssssssss is a subdirectory name you can define with this global default, and fffffff is the form number associated with the object being archived.

GLOBAL TYPE CODES FOR IFS

Global type codes⁵ for managing JAVA, CLASS and HTML files are shipped with TURNOVER® for iSeries. The definitions are identical except for the name and description. You can copy these and modify them as necessary to define your own global IFS type codes.

⁵ Main Menu option 8 (Utility Menu), then option 4.

CHAPTER 3: SETTING UP TURNOVER® FOR ISERIES TO MANAGE IFS OBJECTS

To illustrate, we've shown a sample HTML type code definition, along with a description of the parameters:

8/15/18 08:50:54	Change an IFS Type Code	Your Company, Inc. YOURSYS
Type code	HTML	
Description	HTML files	
Explode	N	
Deploy	N	
Case sensitive	Y	
Sequence	*AFTER	
F3=Exit F6=Add default linked command F7=Select Applications F12=Cancel		

Panel field

Description

Type code	Supply a unique identifier for your IFS type code that is no longer than six alphanumeric characters.
Description	Make it clear what kind of object this type code will be used to manage.
Explode	Type Y if this type of object can be copied, or "exploded," into specific paths defined by your application definition. Type N if this kind of object should not be exploded.
Deploy	Type Y if this type of object should be copied to workstation deployment paths specified in the application's IFS explode/deployment table. Type N if objects of this type should not be deployed.
Case sensitive	Type Y if this type code will be used to promote objects in a case-sensitive file system such as QOpenSys; otherwise, use N .
Sequence	Sequence for IFS type codes has two parts: IFS object sequence and form object sequence. The three-digit number parameter controls how IFS objects are processed in relation to each other. The *BEFORE and *AFTER values control the order in which IFS objects are processed in relation to any other non-IFS objects on the form.

CHAPTER 3: SETTING UP TURNOVER® FOR iSERIES TO MANAGE IFS OBJECTS

Use **F6** to link to this type code a default command that will always be run when promoting objects of this type. For example, if you are promoting a JAVA class object that runs on an IBM i web server, you can optimize it by attaching the IBM i CRTJVAPGM command to the type code you'll use to promote the object. (**Note:** a CLASS type code is shipped pre-configured for this capability.) Read *Optimizing Java Programs* for more information.)

To manage IFS objects, your application needs to know where to archive them; where to get them from and where to put them at each application level; how to assign authority to newly created objects at each level; and how to handle them at promotion time. Certain pieces of the TURNOVER® for iSeries application definition have been changed to accommodate this information.

APPLICATION DEFAULTS AND RULES

If your TURNOVER® for iSeries application includes IFS type codes, an additional field called **IFS Archive subdirectory** appears on the **Application Defaults and Rules** panel:

8/15/18 14:34:53	Application Defaults and Rules Application: DOC Rel: 5 Ver: 1 Lev: 1	Your Company SYSTEM: YOURSYS
Type choices, press Enter.		Field attributes R=Restrict, L=Lock
Source archive file	ARCHIVE	
Library	*DEFAULT	
Archive objects	Y Y,N L	
IFS Archive subdirectory	*DEFAULT	
Field mapping default for copy file command	*MAP *DROP	
Compress out deleted records on copy file command	*YES *YES, *NO	
Require source checkout	Y Y,N	
Checkout error severity	E W,E,blank	
X-Ref method	*NONE *TURNOVER, *HAWKEYE, *ABSTRACT, *NONE	
F3=Exit F12=Cancel		

Panel field

IFS Archive
subdirectory

Description

This is the same as the field specified in TURNOVER® for iSeries Global Defaults.

Here you can override the TURNOVER® for iSeries Global Default⁶ by specifying a subdirectory name where TURNOVER® for iSeries should archive IFS objects managed by this application. To retain the global default value, type *DEFAULT.

The remaining fields on the panel have not changed.

⁶ Main Menu option 8, then option 5.

IFS Path defaults

Plain vanilla IBM i objects live in libraries. TURNOVER® for iSeries uses the **Library Defaults** portion of the application definition to figure out where to get objects from and where to put them during a form run.

IFS objects are stored in directories rather than libraries. A directory that lives within another directory is called a subdirectory. If you want TURNOVER® for iSeries to find the IFS objects you are managing, you have to give TURNOVER® for iSeries directions to the subdirectories where your IFS objects reside. You do that using the **Path Defaults** panel. Find **Path Defaults** by pressing **F8=Path Defaults** from **Library Defaults**:

8/15/18	Path Defaults	Your Company
17:01:38	Application: DOC Rel: 5 Ver: 1 Lev: 1	SYSTEM: SLS
Type choices, press Enter.		
Target path . . . /Docs/Turnover/Review		
		F22=Full path Locked: N
From path /Docs/Turnover/Work		
		F22=Full path Locked: N
F3=Exit F12=Cancel F19=Explode/deploy paths F22=Full path		

A forward slash (/) is used to separate directory names.

As with non-IFS objects, you can “explode” IFS objects into other directories using **F19**. In addition, explode paths for IBM i IFS objects can have a different authority list than the target promotion path.

IFS Line defaults

The IFS equivalent of the **Line Defaults** panel tells TURNOVER® for iSeries how to create the new IFS object in the target directory (copy or move), and whether to delete the old object during the course of the form promotion. (Make sure you read the section about IFS object authority on page 3.)

CHAPTER 3: SETTING UP TURNOVER® FOR iSERIES TO MANAGE IFS OBJECTS

8/15/18 16:36:53	IFS Line Defaults Application: DOC Rel: 5 Ver: 1 Lev: 1	Your Company SYSTEM: SLS
Type choices, press Enter.		
Promotion method	<u>CPY</u>	CPY, MOV, *NULL
Locked	<u>N</u>	Y, N
Target authority list	<u>DOCUSE</u>	Name, F4=List
Locked	<u>N</u>	Y, N
Delete from object	<u>Y</u>	Y, N
Locked	<u>N</u>	Y, N
Development authority list . . .	<u>DOCDEV</u>	Name, F4=List
F3=Exit F4=Select authority list F12=Cancel		

Building TURNOVER® for iSeries authority lists

To control the assignment of authority to the objects it creates, TURNOVER® for iSeries uses an internal object called an *authority list*. (Don't confuse this with the IFS object called an *authorization list*. An authorization list is just one piece of a TURNOVER® for iSeries authority list.)

A TURNOVER® for iSeries authority list is a container in which you can collect:

- Individual user profile information
- IFS authorization list information, and
- Primary group information

all in one place, to use for assigning authority when you promote IFS objects. (Make sure you read the overview of IFS authority on page 3.

The first time you set up an application to use IFS, you will have to create one or more authority lists. You might set up an authority list that sets full access authorities for development, and one that limits access for the non-development levels of your application.

To create and maintain TURNOVER® for iSeries authority lists, position the cursor on one of the authority list fields and press **F4**:

CHAPTER 3: SETTING UP TURNOVER® FOR ISERIES TO MANAGE IFS OBJECTS

We ship a SAMPLE authority list. Let's copy it; then we'll edit it using option 2 to explain the pieces of an authority list:

```

8/15/18      Work with IFS Authority List Objects      T51QA
13:27:19      SLS

Position to list . . .

1=Select 2=Change 3=Copy 4=Delete 5=Display

List      Description      Owner
- DOCDEV   Development authority for documentation  *USRPRF
- DOCUSE   QA, Production auth. for documentation  *USRPRF
- HRPDRD   Human Resources PRD                    QPGMR
- HRQA     Human Resources QA                      QPGMR
- MYNEWLIST Sample IFS Authority List                  *USRPRF
- SAMPLE   Sample IFS Authority List                  *USRPRF

F3=Exit  F6=Add

Copy IFS Authority List Object (TCPYIFSAUT)

Type choices, press Enter.

From IFS authority list . . . . . > SAMPLE      IFS authority list name
To IFS authority list . . . . . mynewlist        IFS authority list name

8/15/18      Work with IFS Authority List Objects      T51QA
13:27:19      SLS

Position to list . . .

1=Select 2=Change 3=Copy 4=Delete 5=Display

List      Description      Owner
- DOCDEV   Development authority for documentation  *USRPRF
- DOCUSE   QA, Production auth. for documentation  *USRPRF
- HRPDRD   Human Resources PRD                    QPGMR
- HRQA     Human Resources QA                      QPGMR
- 2 MYNEWLIST Sample IFS Authority List                  *USRPRF
- SAMPLE   Sample IFS Authority List                  *USRPRF

F3=Exit  F4=Prom  F12=Cancel
F24=More keys

Bottom

8/15/18      Change IFS Authority List Object      Your Company
18:17:40

Type choices and press Enter.

Authority list object . . . . . MYNEWLIST
Description . . . . . Development level authority list for IFS
Owner . . . . . *USRPRF      Name, *USRPRF
Authorization list . . . . . *NONE      Name, *NONE

Data authorities . . . . . *R      *NONE, *RWX, *RX, *RW,...
Object authorities . . . . . *OBJREF      *NONE, *ALL, *OBJEXIST,...

Primary group . . . . . *NONE      Name, *NONE
Primary group data authorities . . . . *RWX      *NONE, *RWX, *RX, *RW,...
Primary group object authorities . . . *ALL      *NONE, *ALL, *OBJEXIST,...

F3=Exit  F12=Cancel
    
```

CHAPTER 3: SETTING UP TURNOVER® FOR iSERIES TO MANAGE IFS OBJECTS

<u>Panel field</u>	<u>Description</u>
Authority list object	The name you assigned to the list appears here.
UNICOM Systems, Inc. Recommends Avoid using blanks in your authority list name. An authority list name that contains blanks can cause a "list or expression not valid" error during Create Missing Items processing for IFS applications.	
Description	Explain what kind of authority this list assigns. For example, does it control development access, testing level, or end user access to the IFS objects for this application?
Owner	Who should own the objects managed by this application? Type a specific user profile, or type *USRPRF to assign ownership to the user who creates the object.
Authorization list	Optional. If an existing IFS authorization list assigns appropriate authority to the appropriate users, you can specify it here. (If none exists, you can create one if you have proper authority. ⁷) Or, you can specify *NONE, and assign authorities using the remaining fields on the panel.
Object authorities	Specifies *PUBLIC ability to manage the existence of this application's IFS objects. See online help for a complete description of parameters you can supply.
Primary group	Identifies a group user profile as the primary group for the IFS objects associated with this application. A group user profile is created as a means of easing the task of assigning authorities on systems where there are large numbers of users.
Primary group data authorities	Defines the data access authority that should be granted to the users in the primary group when new objects are created for this application. See online help for a complete description of parameters you can supply.
Primary group object authorities	Defines the object authority that should be granted to the users in the primary group when new objects are created for this application. See online help for a complete description of parameters you can supply.

⁷ An authorized user can use the WRKAUTL command on a command line.

IFS Application type codes

Once you include a global IFS type code in your application definition, it becomes an application type code with its own application-based default values. The application type code is used to override (and, in some cases, lock) application default values from other areas of the application definition at the object level, giving you more control over what happens to different kinds of objects at promotion time.

This is what an IFS application type code definition panel looks like:

8/15/18 14:49:40	IFSDOC Defaults		Your Company, Inc. SYSTEM: YOURSYS
Application: DOC Rel: 5 Ver: 1 Lev: 1			
Type choices and press Enter.			
Target path . . . /DOCS/TURNOVER/REVIEW		F22=Full path	Locked: <u>N</u>
From path /DOCS/TURNOVER/WORK		F22=Full path	Locked: <u>N</u>
Promotion method <u>CPY</u> CPY, MOV, *NULL			
Locked <u>N</u> Y, N			
Authority list		F4=List	
Locked <u>N</u> Y, N			
Delete from object <u>Y</u>		Y, N	
Locked <u>N</u> Y, N			
Archive <u>Y</u>		Y, N	
Locked <u>L</u> Y, N			
Explode <u>Y</u>		Y, N	Deploy . . . <u>N</u> Y, N
Distribute <u>Y</u>		Y, N	
Development authority list <u>DOCDEV</u>		F4=List	
F3=Exit F4=Select authority list F12=Cancel F22=Full path			

Complete online help is available for this panel.

SUMMARY

Having read this chapter, you should be able to do the following:

- Identify the TURNOVER® for iSeries global defaults that support IFS object management, and find information to help you set them appropriately.
- Locate the sample global type codes for IFS that are shipped with TURNOVER® for iSeries, understand what they do, and successfully copy or modify them for your own use.
- Identify the new pieces of a TURNOVER® for iSeries application definition that relate to managing IFS objects, and successfully create or modify a TURNOVER® for iSeries application definition for that purpose.

If any of these concepts are unclear, reread the appropriate topic and help text. If you still have questions, contact your local Technical Support Representative as outlined under the [SoftLanding website](#).

CHAPTER 4: HOW TO MANAGE IFS OBJECTS

UNICOM Systems, Inc. Note:

These instructions assume that you are using an IBM i system to manage IFS objects. However, we strongly recommend using our TURNOVER® for iSeries Client application for managing PC-based files.

Managing IFS objects with TURNOVER® for iSeries is much the same as for any other IBM i object. The process consists of three basic activities:

- Check out the object
- Change the object and verify the change
- Promote the object.

Note that the subset of activities that make up each of the three basic ones can vary, depending on the kind of objects you are managing and how you have things set up. Checkout and promotion are the activities that this chapter supports. Changing IFS objects and testing the changes depends on the kind of object it is and the environment in which it works – which *could* be entirely independent of the change management process.

This chapter first lists the new commands you need to know about, and then describes how you use them to complete these basic object management steps for IFS objects.

SUMMARY OF COMMANDS FOR MANAGING IFS OBJECTS

These are the TURNOVER® for iSeries API commands that support object management activities for IFS objects:

API	Description	Usage
TCHKOUTIFS	Checking out IFS objects	Command line or program; or Checkout Menu option 11 .
TCHKINIFS	Checking in IFS objects	Command line or program; or Checkout Menu option 24 .
TADDIFSLIN	Adding IFS objects to forms	Program only (includes return parameters).
TADDIFSLNI	Adding IFS objects to forms	Command line or program.
TADDIFSCMD	Linking commands to IFS form lines	Program only (includes return parameters).
TADDIFSCMI	Linking commands to IFS form lines	Command line or program.

The rest of this chapter describes how these commands relate to the three basic object management activities.

CHAPTER 4: HOW TO MANAGE IFS OBJECTS

CHECKING OUT IFS OBJECTS (TCHKOUTIFS)

To check out IFS objects, use option **11** on the Check-out/Check-in Menu⁸. (You can also type the TCHKOUTIFS command on a command line.)

```

Check Out IFS Object (TCHKOUTIFS)

Type choices, press Enter.

Object . . . . .

Type code . . . . . CLASS, DLL, EXE, HLP, HTML,...
Application . . . . . Character value
Reference . . . . . Character value

More...

(Page 2)

Additional Parameters

Path . . . . . *APPL

Copy option . . . . . *ADD *NONE, *ADD, *REPLACE
Release . . . . . *HIGHEST Number, *HIGHEST
Version . . . . . *HIGHEST Number, *HIGHEST
Level . . . . . *PROD Number, *PROD
Programmer . . . . . *USRPRF Name, *USRPRF

More...

(Page 3)

Development path . . . . . *APPL

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
    
```

These additional parameters appear when you press **F10**.

All command parameters are fully explained in the online help.

Use this panel to identify the IFS object you want to check out, supplying information about where the object can be found, where to copy it, its object type, application and reference information, and so on. When you press **Enter**, a checkout record is written to the IFS checkout file and the object is copied according to your instructions.

⁸ Reach the Check-out/check-in source Menu using Main Menu option 5.

Checking In IFS Objects (TCHKINIFS)

Normally, check-in occurs automatically when you run a form. You use this command to check in objects manually only if you checked it out by accident, or if you've realized that an object you checked out doesn't require change.

Check in IFS objects by using option **24** on the Check-out/Check-in Menu⁹. (You can also type the TCHKINIFS command on a command line.)

8/15/18		Check in IFS Objects		Your Company	
Position to object:		Filter by Programmer:			
		Reference:			
		Appl/Rel/Ver/Lev:			
1=Checkin 2=Checkin, delete test object					
Object	Type	Programmer	Reference		
— AUTOMARK.DOC	IFSDOC	MARYS	PROJ0003		
Path: /DOCS/TURNOVER/PUBLISH/R51					
Application: DOC Release: 05 Version: 01 Level: 02					
— DISTRIB.DOC	IFSDOC	MARYS	PROJ0003		
Path: /DOCS/TURNOVER/PUBLISH/R51					
Application: DOC Release: 05 Version: 01 Level: 02					
— EXITS.DOC	IFSDOC	MARYS	PROJ0003		
Path: /DOCS/TURNOVER/PUBLISH/R51					
Application: DOC Release: 05 Version: 01 Level: 02					
— FILES.DOC	IFSDOC	MARYS	PROJ0003		
Path: /DOCS/TURNOVER/PUBLISH/R51					
Application: DOC Release: 05 Version: 01 Level: 02					
+ Note: Objects being promoted on a TURNOVER® for iSeries form are checked in automatically. Do not check in manually.					
F3=Exit		F22=Full path/object			

All command parameters are fully explained in the online help.

This panel shows a list of IFS objects that are checked out. You can limit what you see in the list by filtering for a specific programmer, reference, and application code, release, version and/or level. Select the object you want to check back in with the appropriate option and press **Enter**.

If the **Path** is not completely visible, use **F22** to see it in full.

⁹ Reach the Check-out/check-in source Menu using Main Menu option 5.

CHAPTER 4: HOW TO MANAGE IFS OBJECTS

CHANGING IFS OBJECTS

You cannot edit IFS objects from within TURNOVER® for iSeries.

Once your IFS objects are checked out, you are free to edit them in the target directory, using whatever editing program is appropriate for the kind of object you're working with.

You might also do some compiling, testing, or reviewing at this stage before promoting into a QA or production directory.

PROMOTING IFS OBJECTS

Once you've changed your IFS object(s), you're ready to promote them into your testing directories. The vehicle you use to promote objects is the TURNOVER® for iSeries form. Promoting objects consists of two basic efforts:

1. Creating the form itself.
2. Adding IFS objects to the form.

Summary of IFS functions in *Form Maintenance*

Before you go on to create your IFS form, you should familiarize yourself with the IFS functions that have been added to *Form Maintenance* for that purpose. Here is a summary of the new IFS-related function in *Form Maintenance*:

Function description	Implementation	Notes
Toggle in and out of IFS mode	F8	n/a
Add an IFS form line (in IFS mode)	F6	Uses TADDIFSLIN command
Link commands to IFS form lines (in IFS mode)	Select an IFS line with Option 11 or 12	Uses TADDIFSCMD command
List checked out IFS objects (in IFS mode)	F19	n/a

Creating a form

Easiest: There are a number of ways to create a form. From an IFS standpoint, the most expedient way is to use Main Menu option **6** to start *Work with Forms*, then press **F6** to add a form. Supply a task reference and any other relevant information, and press **Enter**. During this process, TURNOVER® for iSeries looks for objects checked out under the task reference you specify and presents you with two lists of checkout references – one for non-IFS objects (if any), and one for IFS objects. You can use these lists to select the items you want to add to the form. (Read the next topic.)

More complex: There are some other options you might want to consider, depending on your development environment:

- *Command line:* Issue the ADDFORMI command to build a form. (Prompt for online help if needed.)
- *Program:* To have another application create the form automatically, you can write a program that issues the ADDFORM or ADDFORMI command with the correct parameters. (If you use ADDFORMI, your program can run from a command line *or* from within another application program.)

Once the form is created, you have to add to it one line for each IFS object you want to promote. To learn about how to do this from a command line or in a program, read the next topic.

Adding IFS objects to your form (TADDIFSLIN and TADDIFSLNI)

Easiest: If you created your form using **F6** from Work with Forms, then the process of adding lines is automated based on the task reference you supplied for your form. When TURNOVER® for iSeries presents you with lists of checked out objects, you select the ones you want on the form and press **Enter**. The form is built and you can go on to link commands to specific form lines if you so desire (read *Linking pre-run and post-run commands to IFS form lines*).

This is the *Work with IFS Checkout References* panel that appears during the form build procedure, or when you press **F19** in IFS mode of *Form Maintenance*:

```

8/15/18          Work with IFS Checkout References          Your Company
16:49:42

      Form . . . 5103387 Update documentation for IFS.
      Reference . . . PROJ0003

The objects listed below have been checked out using the above
reference number but have not yet been included on the above form.

1=Include on form (To display full path, put cursor on object and press F22.)

Type      Object
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/IFSMAN.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/TROUBLE.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/SETUP.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/INFO.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/FILES.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/EXITS.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/DISTRI.B.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/AUTOMARK.DOC
1 IFSDOC   /DOCS/TURNOVER/PUBLISH/R51/OVERVIEW.DOC

F3=Exit  F11=Show this level  F12=Cancel  F21=Select all  F22=Full path

```

*Complete online
help is available.*

Note: If your form contains both IFS and non-IFS objects, then you see a checkout references panel for each set during the **F6** form build procedure. The panel for regular objects appears first, followed by the IFS objects panel.

More complex: If you're using the command line or program approach, you have these options for adding lines to your form:

CHAPTER 4: HOW TO MANAGE IFS OBJECTS

- *Command line:* Issue the TADDIFSLNI command to add IFS lines one at a time. (This would be the same as pressing **F6** in the IFS mode of *Form Maintenance*.)
- *Program:* To add multiple items at once, your program can issue the TADDIFSLIN or TADDIFSLNI command for each object you want to add. (If you use TADDIFSLNI, your program can run from a command line *or* from within another application program.)

Linking pre-run and post-run commands to IFS form lines

With IFS objects, just as with other kinds of objects, you can issue commands during a form run to further automate your object management activities or customize them in response to specific situations.

There are several ways to do this:

1. From Form Maintenance;
2. Interactively from a command line; OR
3. Globally by type code.

The first two are described next. For the type code method, read *Appendix C: Optimizing Java Programs*.

Most often, you'll do this from Form Maintenance as you add IFS lines:

```

8/15/18                      Form Maintenance - IFS                      Your Company
16:30:13
Form . . . . . 5103387      Update documentation for IFS.
Application . . . . . DOC      Rel/Ver/Lev . . . . . 5/01/01
Job description . . . QBATCH  Output queue . . . . . MARYSOQ
Library . . . . . QGPL      Library . . . . . MARYS
Library list . . . . . DOC050101  Job accounting . . . . .
Programmer . . . . . MARYS      Reference . . . . . PROJ0003
Form comments: N Lock checks: N Completion messages: Y
Position to line . . . . . or Object . . . . .
2=Change 4=Delete          M=Move B=Before A=After 11=PreRunCmd 12=PostRunCmd

Ln Object                      Type      Cd  Path
11 001 HOWTO.DOC              IFSDOC    A  /DOCS/TURNOVER/REVIEW
            
```

*All command
parameters are
fully explained in
the online help.*

```

8/15/18                      Add IFS-Linked Command                      Your Company
16:38:31

Form . . . . . 5103387
Command type . . . Pre-Run
Sequence . . . . . 1

Type command, press Enter.

Command to execute: type your command here...

Error recovery: C  C=Cancel (end job)  G=Go (continue job)

Link to line: 1      Object: HOWTO.DOC
              F10=Select      Path: /DOCS/TURNOVER/REVIEW
                              Type: IFSDOC

Library list . . . . *CURRENT *APPL, *CURRENT

F3=Exit F4=Prompt F9=Retrieve F10=Select F12=Cancel
            
```

This would be the same as prompting the TADDIFSCMI command and pressing **F10** to see all the parameters.

Commands linked to IFS form lines run either immediately before or immediately after the IFS object is processed.

About writing programs for IFS-related form processing

If you are dealing with a large number of objects, and want to write a program that issues the ADDFORM, TADDIFSLIN or TADDIFSCMD command to add lines and link commands, there are some things you should keep in mind.

- These commands use an additional parameter to return information to the calling program.
- You do not have to use the parameter, but your program must define it.
- If you aren't using the additional parameter, make sure your program monitors for and ignores the resulting error message.

CHAPTER 4: HOW TO MANAGE IFS OBJECTS

RUNNING YOUR IFS FORM

On the **Work with Forms** panel, select your form with option 7 to run the form. TURNOVER® for iSeries will process the objects as your application definition and type codes indicate, applying any overrides you've made.

Check the form logs when the form job completes to verify that all went well, and make sure that your IFS objects are where you expected them to be.

SUMMARY

You should now be able to do the following activities:

1. Recognize API commands for IFS and find information about their parameters.
2. Check out IFS objects.
3. Check in IFS objects.
4. Add IFS objects to a form using Form Maintenance or API commands.
5. Link pre- or post-run commands to an IFS form line.
6. Run your IFS form.

If any of these concepts are still unclear, review the necessary topic or contact your local Technical Support representative as outlined under the [SoftLanding website](#).

CHAPTER 5: DISTRIBUTING IFS OBJECTS

Distributing IFS objects is much like distributing other IBM i objects, except it's done with directories rather than libraries.

DISTRIBUTION-RELATED SUBDIRECTORIES FOR IFS OBJECTS

The first time you run a promotion that includes distributing IFS objects, TURNOVER® for iSeries looks for an IFS directory called /TURNOVER. (If such a directory doesn't already exist, TURNOVER® for iSeries creates it.) Within the /TURNOVER directory, TURNOVER® for iSeries creates these subdirectories for current and future use during IFS distribution processing:

/DISTRIBUTION	This is the distribution work directory. When a distribution job runs that includes IFS objects, the following subdirectories are created within the work directory:
/DISnnnnnnn	Named for the distribution job associated with the form.
/ffffff	Named for the form number containing IFS objects for distribution.

Distributed objects are created in the /ffffff subdirectory as lll, where lll is the object's form line number.

For example, the IFS object described by IFS line 4 of form 5103389, which ran under distribution job DIS007252, would appear in the IFS distribution directory as:

/TURNOVER/DISTRIBUTION/DIS007252/5103389/004

/RECEIVE	This is the receive form directory. This directory is created on remote computers where a production computer has sent one or more forms for installing. When the remote computer receives a form that contains IFS objects, a subdirectory and objects are created here as follows:
/ffffff	Named for the form number containing the IFS objects that were distributed to the remote computer.

Received objects are created and named lll, where lll = the IFS line number.

For example, the IFS object described by IFS line 4 of form 5103389 received from the production computer would appear in the IFS receive directory as:

/TURNOVER/RECEIVE/5103389/004

OTHER PREPARATIONS FOR DISTRIBUTING IFS OBJECTS

- Make sure the appropriate IFS application definition(s) have been created on your remote computers. (Try out the Remote Configuration Wizard – Distribution Menu option 7.)
- Make sure your application's IFS type codes are set up to distribute objects, as desire

CHAPTER 6: FINDING TURNOVER® FOR ISERIES INFORMATION ABOUT IFS OBJECTS

TURNOVER® for iSeries keeps records of the activities it performs on any object, and IFS objects are no exception. Various kinds of information are available for when you need to track the promotion, archive, distribution, or checkout history for an IFS object. In addition, information about checked out IFS objects is available from the Source Check-in/Checkout Menu.

IFS OBJECT HISTORY

To see a list of IFS objects TURNOVER® for iSeries has promoted, use Main Menu option 4; then press **F8**:

```

8/15/18                                Work with IFS Object History          T51QA
10:54:46                                SLS

To reposition screen, enter Object:
Filter by Type:                        Programmer:                        Appl:
Status:                               Reference:

3=Copy archived object  5=View form  6=Explode paths  8=Archive information
9=View task  10=Checkout history  11=Line details  16=Dist History

Object      Type      Path      Cd Status      Run Date
--- AUTOMARK.DOC      IFSDOC      ...RNOVER/REVIEW A  Ready      0/00/00
      Reference: PROJ0003  Pgm: MARYS      Archived: N Checked out: N
      Form: 5103389 Appl: DOC Rel: 5 Ver: 1 Lev: 1
--- DISTRIB.DOC      IFSDOC      ...RNOVER/REVIEW A  Ready      0/00/00
      Reference: PROJ0003  Pgm: MARYS      Archived: N Checked out: N
      Form: 5103389 Appl: DOC Rel: 5 Ver: 1 Lev: 1
--- EXITS.DOC      IFSDOC      ...RNOVER/REVIEW A  Ready      0/00/00
      Reference: PROJ0003  Pgm: MARYS      Archived: N Checked out: N
      Form: 5103389 Appl: DOC Rel: 5 Ver: 1 Lev: 1
More...

F3=Exit  F4=List      F11=Show all      F12=Cancel
F21=System command  F22=Full path

```

From this panel, you can see and filter a list of all IFS objects that TURNOVER® for iSeries has managed. The most recent information for each object appears, and the list includes objects only for those applications you're authorized to use.

You can select any object in the list with any applicable option. Descriptions of some of the more relevant options follow; for more information, read the online help for this panel.

Archive history

If an IFS object was archived during the course of its promotion, this panel shows you that (**Archived** label). To find out when and where an object was archived, select it with option 8.

Archive subdirectories for IFS objects

TURNOVER® for iSeries stores archived objects in the /TURNOVER/ARCHIVE subdirectory. If this directory doesn't exist, TURNOVER® for iSeries creates it.

CHAPTER 6: FINDING TURNOVER® FOR ISERIES INFORMATION ABOUT IFS OBJECTS

In addition, you can specify in your application definition a default archive subdirectory that your application should use when archiving objects. This value is set using the TURNOVER® for iSeries global default ***IFS Archive subdirectory***, and is shipped set to DEFAULT. You can override the global default for any level of your application by specifying a different value on the ***Application defaults and rules*** panel for that application level.

Archiving capabilities

When you use option **8** to view archive information about an object, this panel appears:

8/15/18		Work with Archived IFS Objects			T51QA	
Object . . . DISTRIB.DOC						
Path /DOCS/TURNOVER/REVIEW						
1=Restore 3=Copy						
Form	Line	Archive Date	Online	Save Date	Save Volume	
5103391	001	12/12/98	Y			
5103391	002	12/12/98	Y			
5103214	001	03/08/98	N	04/15/98	SLS210	
5103214	002	03/08/98	N	04/15/98	SLS210	
5103220	002	05/19/98	Y	07/15/98	SLS211	
F3=Exit F12=Cancel						

An object can be saved offline and still exist online. Any object that exists online is eligible for the ***Copy*** option, whether it's been saved offline or not. An object that has been saved offline and no longer exists online is eligible only for the ***Restore*** option.

TURNOVER® for iSeries Managing IFS Objects with TURNOVER® for iSeries CHAPTER 6: FINDING TURNOVER® FOR ISERIES INFORMATION ABOUT IFS OBJECTS

Checkout history

Checkout history shows any applicable checkout and checkin dates and times for any IFS object that you select with option **10** on the *Work with IFS Object History* panel:

```

8/15/18          Display Checkout History          T51QA
Object . . . OVERVIEW.DOC

Path . . . . /DOCS/TURNOVER/REVIEW

Position to Date . . . . _____ Filter by Programmer . . . _____
Reference . . . _____

Development Path      Check-out      Check-in      Programmer Reference
/DOCS/TURNOVER/WORK  12/03/98 14:18  12/04/98  8:29  MARYS      PROJ0003

F3=Exit  F12=Cancel  F22=Full path
    
```

Distribution history

If you are distributing IFS objects during promotions, then selecting an IFS object with option **16** on *Work with IFS Object History* shows you a list of systems where it has been sent. Relevant job information is included:

```

8/15/18          Maintain Form Distribution Status          T51QA
12:42:59

Form . . . . . 5103375
Description . . . IFS Documentation Development
Status . . . . . RAN-OK

2=Change  8=Work w/ Distribution Job

Sys System  Description      Distribution
- 150 WASHDC  SLS Washington DC  DIS5100074 05/19/98  Y  *SNADS  Y
- 410 HOUSTON SLS Houston      DIS5100074 05/19/98  Y  *SNADS  Y
- 420 SEATTLE SLS Seattle      DIS5100074 05/19/98  Y  *SNADS  Y
- 430 ATLANTA SLS Atlanta      DIS5100074 05/19/98  Y  *SNADS  Y

F3=Exit  F12=Cancel
    
```

This panel is the same regardless of whether you are looking at IFS or non-IFS objects.

CHAPTER 6: FINDING TURNOVER® FOR ISERIES INFORMATION ABOUT IFS OBJECTS

CHECKED OUT IFS OBJECTS

Another source of information about IFS objects is on the Checkout/Checkin Source Menu¹⁰. When you take option **36**, you see a list of all IFS objects that are checked out:

```

8/15/18          View Checked Out IFS Objects          T51QA

Position to object: _____ Filter by Programmer: MARYS
                                   Reference: _____
                                   Appl/Rel/Ver/Lev: _____

2=Change checkout record

Object                                     Type      Programmer Reference
- AUTOMARK.DOC                             IFSDOC     MARYS      PROJ0003
    Path: /DOCS/TURNOVER/PUBLISH/R51
    Application: DOC Release: 05 Version: 01 Level: 02 Exclusive: Y
- DISTRIB.DOC                             IFSDOC     MARYS      PROJ0003
    Path: /DOCS/TURNOVER/PUBLISH/R51
    Application: DOC Release: 05 Version: 01 Level: 02 Exclusive: Y
- EXITS.DOC                               IFSDOC     MARYS      PROJ0003
    Path: /DOCS/TURNOVER/PUBLISH/R51
    Application: DOC Release: 05 Version: 01 Level: 02 Exclusive: Y
2 FILES.DOC                               IFSDOC     MARYS      PROJ0003
    Path: /DOCS/TURNOVER/PUBLISH/R51
    Application: DOC Release: 05 Version: 01 Level: 02 Exclusive: +

F3=Exit  F4=Reference selection  F12=Cancel  F22=Full path/object
    
```

Here you can review a complete list of IFS objects that are currently checked out.

As with non-IFS objects, option **2=Change checkout record** is handy if you are coordinating development activity between multiple programmers or across multiple projects and tasks. You can also use it to change the type code if you've checked out the object under the wrong type code and want to correct that without checking the object in and then out again.

```

8/15/18          Change Checkout Record          T51QA
12:58:39      Application: DOC Rel: 5 Ver: 1 Lev: 2

Object . . . . . FILES.DOC

Type code . . . . . IFSDOC
Checked out from path . . . /DOCS/TURNOVER/PUBLISH/R51

Development path . . . . . /DOCS/TURNOVER/WORK

Programmer . . . . . MARYS
Reference . . . . . PROJ0003

New type code . . . . . *SAME          *SAME, Type code, F4=List
New programmer . . . . . *SAME          *SAME, *USRPRF, Name, F4=List
New reference . . . . . *SAME          *SAME, Char, F4=List
                        *SAME

Copy test object . . . . . N          Y,N
Delete test object . . . . . N          Y,N

F3=Exit  F4=List  F12=Cancel
    
```

¹⁰ Main Menu option 5.

SUMMARY

After reading this chapter, you should be able to do the following:

1. Access IFS object history to find information about the IFS objects you are managing with TURNOVER® for iSeries.
2. Locate and manage archived IFS objects using the **Copy** and **Restore** options appropriately.
3. Locate and review the checkout history of an IFS object.
4. Locate distribution history information for an IFS object, and manually change the distribution status of an IFS object for a remote system; as well as display information about the associated distribution job.
5. Locate a list of checked out IFS objects using the Checkout/checkin Source Menu, and be able to change the checkout record for an IFS object.

If any of this information is unclear, please reread the topic. If you still have questions, contact your local Technical Support representative as outlined under the [SoftLanding website](#).

APPENDIX A: IFS FILE SUMMARY

The following files are used to record TURNOVER® for iSeries information about IFS objects:

File Name	Description
TIFSEXTF	IFS Path Extension file
TIFSDFTF	IFS Application Defaults file
TIFSAUTF	IFS Authority List Object file
TIFSTYPF	IFS Application Type Code file
TIFSEXPf	IFS Application Explode List
TIFSOUTF	IFS Checkout file
TIFSLINF	IFS Form Line file
TIFSCMDF	IFS Linked Commands file
TIFSARCF	IFS Archive Index file
TIFSOFFf	IFS Offline Archive Index file
TIFSCONF	IFS Conflict file

IFS EXIT POINTS

This appendix describes several exit points that are provided so you can customize TURNOVER® for iSeries' IFS processing to suit your needs. To view the exits that are provided for IFS, use Main Menu option **8**, then option **7** to reach Work with Exits. Then, press **F8** to see the list of IFS exit points.

To view online help for any exit, select it with option **8**. Sample program source code members that declare all expected parameters are also provided for each exit. To browse a sample member, select the associated exit with option **7**, then choose the associated source member with an option to browse, print, or copy.

UNICOM Systems, Inc. Recommends

As with all sample code UNICOM Systems, Inc. provides, we recommend copying the sample source before modifying it to avoid loss of the original.

Cross-reference information is not currently supported for IFS objects. However, you might use one or more of these exit points to perform some of your own cross-referencing.

IFS 2: POST CHECKOUT EXIT

Exit IFS 2 is called after an IFS object has been checked out and copied to your development path.

Example

This exit can come in handy when working with IFS objects that are associated with supplemental objects such as control files or help files. In this situation, the exit program can check the IFS object name and, if it has dependent files, automatically copy them to the development path. You might also use your program to raise a flag or send a message when company sensitive objects are checked out.

IFS 3: MANUAL CHECK-IN EXIT

Exit IFS 3 is called when you manually check in an IFS object. Use this exit to perform additional check-in security processing, and/or other custom processing.

With manual check-in, the specified IFS object does not replace the production object. Manual check-in simply frees the lock on the object so that someone else can check it out.

When a user tries to check in an IFS object manually, TURNOVER® for iSeries will verify that the user has check-in authority in general and for the particular application. If the user does not have check-in authority, then TURNOVER® for iSeries will not allow the object to be checked in.

Example 1

Depending on the programming environment in your company, you may want to perform a more extensive verification procedure. For instance, you may want to allow check-in only by the same programmer who checked out the object. Furthermore, you may want to allow that programmer to check in only objects that have a particular attribute. For example, perhaps programmer A is only allowed to check in HTML objects that he/she previously checked out.

Example 2

In addition to security screening, you can use IFS Exit 3 to perform other tasks when an object is checked in. For instance, perhaps a particular object is associated with control or help files that are generally checked out with the object. In this case, your IFS Exit 3 program could search the path for these files, and if they exist, automatically check them in as well.

IFS 4: ADD LINE EXIT

Exit IFS 4 is called whenever an IFS line is added to a form – whether implicitly or explicitly. Because of this, you might decide to change or evaluate one or more of the many parameters passed in to the exit program.

Any of the parameter values changed within the exit program take precedence over those defined during the application default process.

Example

Suppose that when an IFS line is added to a form, you want to check the associated action code. If it happens to be a D, indicating object deletion, you may want to send a message to the user(s) on the form approval list, as well as to QSECOFR, warning them of the pending deletion.

IFS 5: PRE-CHECKOUT EXIT

Exit IFS 5 is called before IFS objects are checked out and copied to your development path. Given the list of parameters that are passed to the exit program, there are many different conditions you can evaluate to determine whether or not to allow a checkout request to proceed.

The last parameter passed, &CHECKOUT, is a flag that can be modified by your IFS Exit 5 program. Although initialized to Y by the calling program, if one of your conditions for checkout fails, you can set this flag to N to cancel the checkout request.

As part of the standard TURNOVER® for iSeries checkout process, these conditions are already evaluated:

- TURNOVER® for iSeries verifies that the user requesting a given member/object has general checkout authority, as well as authority for the particular application.

APPENDIX A: IFS FILE SUMMARY

- TURNOVER® for iSeries ensures that the requested object is not already checked out.

You may want to go beyond the standard TURNOVER® for iSeries evaluation process for checkout.

Example

Perhaps, for a given application, you want to allow only users belonging to a specific group profile to check out an object.

Note: For parameter &CHECKOUT: Y = success; N = failure.

IFS 7: POST ADD LINE EXIT

Exit IFS 7 is call after a line is added to a form. If your exit program must change the parameters to the line that is being added, then you should use exit IFS 4 (IFS Add Line Exit) instead of exit IFS 7. However, if your processing requires a TURNOVER® for iSeries line number, you should use exit IFS 7, because the line number is not yet available when exit IFS 4 is called.

Example

You might use this exit if you need to update a database to reflect the form and line number associated with the object being added to the form.

APPENDIX B: TROUBLESHOOTING

This chapter describes some issues you might face as you begin working with IFS objects. Here you can find explanations for situations that can occur, and instructions for correcting them.

Q

I checked an existing IFS object out of the production folder. The confirmation message said "Object checked out." But when I look in my development folder, the object isn't there. Why not?

A

TURNOVER® for iSeries couldn't find the IFS object in the production folder, so it reserved a new object in the specified path and checked it out.

If TURNOVER® for iSeries finds an IFS object in the production folder, it copies it, and you see the message "Object copied." If the object is not found in production, it is considered new, and you see the message "Object checked out."

To solve the problem, determine why TURNOVER® for iSeries could not see your object. Usually, this is caused by a discrepancy in the path you expected or supplied, as opposed to the one specified in the type code for that application level. The application type code remembers the appropriate IFS paths for each level of your application; all you need to supply on the checkout panel is the object name. If you specified the full path for your object on the checkout panel, TURNOVER® for iSeries looked for those directories within the path it already knew about, and that's why your object was not found.

APPENDIX B: TROUBLESHOOTING

Q

My IFS processing failed because of insufficient authority. (This can happen during checkout or form processing.)

A

A TURNOVER® for iSeries Global Default (*IFS Adopted authority user*) identifies the user profile from which TURNOVER® for iSeries should adopt authority for managing IFS objects. If you did not adjust this value when you set up TURNOVER® for iSeries for IFS object management, then it defaults to *CURRENT, which means that the user profile of the user doing the work is used. If that user profile does not have *ALLOBJ authority, then TURNOVER® for iSeries jobs that manage IFS objects to which the user does not have authority will fail.

We recommend creating a user profile that has a valid password and *ALLOBJ authority on your system, and specifying that profile for the *IFS Adopted authority user* parameter in Global Defaults.

If you plan to manage objects on a Windows computer in your network (with the TURNOVER® for iSeries Client), you should also have your Windows Administrator create an identical user and password on your Windows system, and give this user full permission to any network shares you plan to manage using the TURNOVER® for iSeries Client. For more information about managing IFS objects using the TURNOVER® for iSeries Client, read **Appendix E** in this book, along with your **TURNOVER® for iSeries Client Getting Started Guide**.

Q

There are no objects in the /TURNOVER/RECEIVE directory on my remote computer. How can I find out if TURNOVER® for iSeries successfully packaged the IFS objects on my form before distributing?

A

TURNOVER® for iSeries packages IFS objects in their own individual save file, named IFSnnnnnn, where nnnnnn is the job number associated with the distribution job. This save file is in turn packaged in the save file that contains the form and any non-IFS objects associated with it.

Check the distribution joblog on the production system for error messages pertaining to the creation of an IFSnnnnnn save file.

APPENDIX C: OPTIMIZING JAVA PROGRAMS

This appendix describes how to expand the basic IFS support with a special automatic linked command feature that lets you build Java programs from CLASS, JAR, or ZIP files during promotion. If you run Java programs from your IBM i server, this feature might be useful for you.

OVERVIEW

This feature allows you to create and sustain a default linkage between CLASS, ZIP, or JAR object types on your TURNOVER® for iSeries form to the IBM i CRTJVAPGM command. The CRTJVAPGM command builds a hidden IBM i Java program that becomes part of the CLASS, ZIP, or JAR file and runs on the IBM i when the class is started by the IBM i JAVA command.

While you do not have to create the hidden commands when you promote these kinds of files (the IBM i JAVA command will build them on the fly if they don't exist¹¹), doing so can offer you these benefits:

- improved initial runtime for the class
- greater control over special creation parameters (such as USRPRF, USEADPAUT, which are not available on the JAVA command¹²)
- greater control over optimization parameters (as opposed to the default optimization settings used by the JAVA command on the fly).

CONFIGURING DEFAULT LINKED COMMANDS

Before you proceed, you should familiarize yourself with the CRTJVAPGM command and its parameters as they pertain to your version of the IBM i operating system. Understand the values you'll want to specify and study the values that we ship already linked to the CLASS global type code.

To configure TURNOVER® for iSeries for specialized processing of Java files, follow these instructions.

¹¹ Depends on the value of the **Optimization** parameter of the IBM i JAVA command. Read the online help for the JAVA command for more information.

¹² These parameters were added to the CRTJVAPGM command for V4R4M0. If you want to specify values for them you can override the command in the application type code definition, or by editing the form line.

APPENDIX C: OPTIMIZING JAVA PROGRAMS

Step 1: Check your global CLASS (or JAR or ZIP) type code

1. Edit your global type code with option 2.
 - a. If you are a new TURNOVER® for iSeries user, an existing CLASS type code might already have the CRTJVAPGM command linked as a default post-line command. All you have to do is verify the command parameters, adjusting them if necessary.
 - b. If you upgraded from a previous release of TURNOVER® for iSeries, then your type codes were preserved during the upgrade. If you have an IFS type code you want to use, you can edit it and add the default linked command. In edit mode, add the linked command using **F6**:

5/03/00	Add IFS Type Code Default Command		Your Company, Inc.
Type code	<u>CLASS</u>		
Command type . . .	<u>*POSTRUN</u> *PRERUN, *POSTRUN		
Sequence	<u>10</u>		
Type command, press Enter.			
Command to execute: <u>CRTJVAPGM CLSF('&PO') OPTIMIZE(30) REPLACE(*YES)</u>			
&TY=Type code	&FP=From path	&PG=Programmer	&PO=Path/Object
&TP=Target path	&FO=From object	&RF=Reference	&UO-&U9=User defined
&OB=Target object	&ME=Method	&AL=Authority list	
Error recovery: <u>C</u> C=Cancel (end TurnOver job) G=Go (continue TurnOver job)			
Library list <u>*CURRENT</u> *APPL, *CURRENT			
Run on system <u>*ALL</u> *ALL, *LOCAL, *REMOTE, Name			
F3=Exit F4=Prompt F12=Cancel			

- c. If no global IFS type code exists for the purpose, press **F6** on the **Work with Type Codes** panel to add a global type code, and then press **F8** to add an IFS type code. Fill in the necessary information and exit; then edit it and add the command using **F6** as described above.

Step 2: Turn on automatic linking at the application level

Include the global type code in your application (at all levels), and change the application level definition to turn on the automatic include feature by setting the *Add default linked commands* parameter to **Y**:

8/15/18	CLASS Defaults	Your Company, Inc.
08:33:04	Application: WEB Rel: 1 Ver: 1 Lev: 1	SYSTEM: YOURSYS
Type choices and press Enter.		
Target path . . . /website/qa	F22=Full path	Locked: <u>N</u>
From path /website/dev	F22=Full path	Locked: <u>N</u>
Add default linked commands . . <u>Y</u>	Y, N	
Promotion method	<u>CPY</u>	CPY, MOV, *NULL
Locked	<u>N</u>	Y, N
Authority list	<u>WEBPROD</u>	F4=List
Locked	<u>Y</u>	Y, N
Delete from object	<u>N</u>	Y, N
Locked	<u>N</u>	Y, N
Archive	<u>Y</u>	Y, N
Locked	<u>Y</u>	Y, N
Explode	<u>N</u>	Y, N
Distribute	<u>Y</u>	Y, N
Development authority list . . .	<u>WEBDEV</u>	F4=List
F3=Exit	F4=Select authority list	F12=Cancel F22=Full path

VERIFY YOUR SETUP

1. Copy an existing CLASS file that you can use as a test. Store the file in the production directory identified by the CLASS type code at your application's highest level.
2. Locate a valid project and task reference you can use to check out your test class file.
3. Check out the CLASS file using option 11 on the Check-in/Checkout menu, or use the TCHKOUTIFS command from a command line.
4. Add the CLASS file to a form using **F6** from *Work with Forms*.

APPENDIX C: OPTIMIZING JAVA PROGRAMS

5. You should now be looking at the form in *Form Maintenance*:

8/15/18	Form Maintenance - IFS		Your Company
11:22:28			
Form	5300484	Fix broken class file	
Application	WEB	Rel/Ver/Lev	1/01/01
Job description	QPGMR	Output queue	PGMR10Q
Library	QGPL	Library	PGMR1
Library list	WEB010101	Job accounting	
Programmer	PGMR1	Reference	WEB 0003
Form comments: N	Lock checks: Y	Completion messages: N	
Position to line	or Object		
2=Change 4=Delete M=Move B=Before A=After 11=PreRunCmd 12=PostRunCmd			
Ln	Object	Type	Cd Path
001	myclass.class	CLASS	A /web/qa
2	CRTJVAPGM CLSF('&PO')	OPTIMIZE(30)	REPLACE(*YES)
Bottom			
F3=Exit F4=Comments F5=Resequene F6=Add line F12=Cancel F24=More keys			

At this point, you can use option 2 to edit the CRTJVAPGM command and override its parameter values if necessary.

6. Run your form. Monitor the job using option 8 in *Form Maintenance*. Messages issued by the CRTJVAPGM command are written to the TURNOVER® for iSeries Log Report for the form job.

APPENDIX D: PROMOTING FILES IN THE WINDOWS FILE SYSTEM

This chapter describes how to use TURNOVER® for iSeries to manage objects that reside on a Windows-based computer in your network.

It's assumed that you're already familiar with the basic IFS support described by the chapters in this guide.

UNICOM Systems, Inc. Recommends

Be sure to check out UNICOM Systems, Inc.'s TURNOVER® for iSeries Client application, designed for managing e-Business and Web applications using TURNOVER® for iSeries!

OVERVIEW

As of V4R3M0 of the IBM i operating system, IBM added a new file system to the IFS support on the IBM i system. This new file system is called the Windows file system, and it allows a program on your IBM i system to see workstation computers that are connected to the same network domain. This is functionally equivalent to the "Network Neighborhood" that you see in Windows 95/98/NT, and it means that TURNOVER® for iSeries can now manage PC objects directly.

Note: TURNOVER® for iSeries cannot manage objects on a workstation that is running Windows 95/98. However, you can work around this by setting up a shared folder with subdirectories on a Windows server in your network and map the folder as a drive on the Windows 95/98 client.

SYSTEM REQUIREMENTS

- TURNOVER® for iSeries (tape date 2/1/2000 or later), or higher.
- IBM i V4R3M0, with current fixes applied.
- NT 4.0, Windows 2000, or higher, on the client PC.

PREREQUISITES

To access PCs in your Windows network, the IBM i NetServer feature must be configured and started on your IBM i computer. For more information about NetServer, point your web browser at <http://www-03.ibm.com/systems/power/software/i/netserver/>.

APPENDIX D: PROMOTING FILES IN THE WINDOWS FILE SYSTEM

CONFIGURING FOR WINDOWS FILE MANAGEMENT

Before you can manage objects directly on a PC, you must make the following adjustments:

1. The user specified as TURNOVER® for iSeries' *Adopted IFS authority user* (in TURNOVER® for iSeries' Global Defaults) must also exist on your Windows network, with an identical profile name and password. Consult with your Windows Administrator to create this profile as needed.
2. Share the appropriate Windows directories on the client PC, and give full control permissions to the *IFS Adopted authority user* profile on the Windows network. (If you need help with this, contact your Windows Network Administrator.) From Windows, you would access these shared folders from another computer by navigating there through Network Neighborhood or by direct file specification, something like this:

```
\\SystemName\ShareName\SubDir\Filename.ext
```

3. Configure your TURNOVER® for iSeries application definition type codes with the correct path statements.

From the IBM i, you would access the same shared folders using a path statement similar to this:

```
/QNTC/SystemName/ShareName/SubDir/Filename.ext
```

TROUBLESHOOTING WINDOWS SUPPORT

- We recommend limiting the IFS adopted authority user's access to only those folders you want TURNOVER® for iSeries to manage.
- You can use the IBM i command WRKLNK to access workstation computers in the Windows Network Neighborhood, as follows:

```
WRKLNK OBJ ('/QNTC/*')13
```

This command will produce a list of all the workstation computers the IBM i can "see" on the network.

If you've shared the appropriate directories on the computer you want TURNOVER® for iSeries to access, but it doesn't appear in the IBM i computer's list of accessible workstations, then you can make it available by running the IBM i MKDIR command. For example:

```
MKDIR DIR ('/QNTC/WindowsSystemName')
```

...where *WindowsSystemName* is the name of the workstation computer as it appears in the Windows Network Neighborhood. (There might be fixes available from IBM that address this problem.)

¹³ Also accessible from the DATA menu, option 5 (Integrated File System commands), then option 2 (Object commands), then option 1 (Work with object links). Specify '/QNTC/*' for the object parameter and press **Enter**.

As always, we recommend that you run a test using your reconfigured TURNOVER® for iSeries application before you begin managing live files.

APPENDIX E: PROMOTING FILES IN A UNIX FILE SYSTEM

This appendix describes TURNOVER® for iSeries' UNIX file system support.

PREREQUISITES

UNIX support requires TURNOVER® for iSeries, or Release 5.3, with a tape date of 1/9/2002 or later.

SUPPORTED OPERATING SYSTEMS

This should work with any operating system that provides a Network File System (NFS) server, including any version of UNIX. UNICOM Systems, Inc. has tested this with the following operating systems:

- RedHat Linux 7.1
- Mac OS X 10.1.2

(If you try this with another UNIX platform, please send us your results!)

BACKGROUND

TURNOVER® for iSeries can manage files that reside in the Integrated File System (IFS). The IBM i IFS is a channel that allows the IBM i system to recognize objects from other operating system file systems. Some of these file systems physically reside on the IBM i system itself; others reside on other computers that are accessed using networking protocols.

It doesn't matter to the IBM i system (and to the APIs that it provides) where the files physically reside, as long as the IBM i can access them.

Because TURNOVER® for iSeries' Integrated File System support is built on top of the IBM i IFS API's, TURNOVER® for iSeries can access and manipulate any file that the IBM i can manage. The features and limitations of TURNOVER® for iSeries mirror those of the IBM i APIs that it uses.

One file system that the IBM i operating system natively supports is called Network File System (NFS). The NFS has been around for a very long time, and is probably the standard in the UNIX world for networking systems. All versions of UNIX provide support for NFS.

Because NFS is a Client/Server protocol, the IBM i computer on which TURNOVER® for iSeries resides becomes the Client computer, and the UNIX system is the Server.

APPENDIX E: PROMOTING FILES IN A UNIX FILE SYSTEM

CONFIGURING TURNOVER® FOR iSERIES SUPPORT FOR UNIX

Follow these steps to set up TURNOVER® for iSeries' support for UNIX objects.

Step 1: Define Exports on the UNIX System(s)

On the UNIX system you must define the directories that you want to make available to the IBM i system. In UNIX terminology, this is called an “export.”

To share a UNIX directory with other systems, you must export it. On Linux systems, you do this by editing a file named `/etc/exports`¹⁴.

This is a simple text file that contains a list of all of the directories you want to share, as well as some options that control how they are shared, and to whom. Here is an example:

Part 1	Part 2	Part 3
<code>/opt/WebSphere/AppServer/installedApps</code>	<code>as400.mycompany.com</code>	<code>(rw,no_root_squash)</code>
<code>/pub</code>	<code>*(ro,all_squash)</code>	<code># Public dir</code>
<code>/home</code>	<code>*.mycompany.com</code>	<code>(rw,squash uids=0-99) # User's Home directories</code>

In this example, the file contains entries for three directories. For the purposes of TURNOVER® for iSeries and this document, we're concerned only with the first line and its three parts.

Part 1 of the line identifies the directory you're sharing – in this case, `/opt/WebSphere/AppServer/installedApps`. Because this Linux system is running WebSphere, we want to share the directory where our WebSphere applications are installed, so that we can use TURNOVER® for iSeries to manage them.

Part 2 of the line defines the system(s) to which this export should be accessible. In this case, we want to make this export available only to our IBM i system. To identify the system, you can supply a TCP/IP name (as the example shows), one IP address, or a range of IP addresses. For example, a range such as `192.168.0.0/255.255.255.0` would allow any system in the IP address range of `192.168.0.1 - 192.168.0.254` to access the share.

¹⁴ To find out how to share directories on other UNIX platforms, consult your system administration documentation.

Part 3 of the line is important, and it defines some of the sharing options. The value `rw` stands for read-write, and is necessary if TURNOVER® for iSeries is going to be able to install and change files. After that there is value of `no_root_squash`. This value indicates that it is OK for the root user (which is QSECOFR on an IBM i system) to connect. By default, NFS maps the root user to anonymous for security reasons. TURNOVER® for iSeries needs to be able to connect as the root user to apply owner and permission settings, as well as to just guarantee that TURNOVER® for iSeries has the authority necessary to update the files. As long as you feel that your QSECOFR profile is safe on the IBM i system, this is not a major security concern.

Once you have defined your exports, you need to run a command on your UNIX box to "refresh" your exports. On Linux, this command is:

```
/user/sbin/exportfs -a -v
```

The `-a` switch indicates that all directories in the `/etc/exports` file should be exported. The `-v` switch says to print verbose output. This helps troubleshoot syntax problems in your exports file.

Step 2: Verify NFS service and firewall rules

The System Administrator should make sure that the NFS service is running on the UNIX system, and that network firewall rules do not deny access to the service.

Administrators can check the NFS service from the server by typing:

```
ps ax | nfsd
```

A list of `nfsd` server processes should appear. If not, try starting the `nfsd` daemon by typing:

```
/etc/init.d/nfs start
```

If that works, then the administrator can set up NFS to start permanently using the SYSV Runlevel manager (`tkssysv` command).

An administrator can view firewall rules by typing:

```
ipchains -L
```

Only an authorized system administrator should change firewall rules.

Step 3: Mount Directories on IBM i systems

The next step is to "mount" the exported file system on the IBM i system. This makes the files and directories of the UNIX system available to the IBM i. When you mount a file system, you have to mount it on top of an existing directory, which becomes the "mount point."

APPENDIX E: PROMOTING FILES IN A UNIX FILE SYSTEM

All of the contents of that directory disappear and are replaced by the contents from the mounted file system. (When the file system is un-mounted, the contents reappear.)

We recommend creating a directory in the IFS root folder named /NFS, to contain all of your mount points. You can then create subdirectories in /NFS, one for each UNIX system you plan to mount, and further subdirectories for each directory you will be mounting. For example:

```
/NFS
  /Linuxserver
    /WebSphere
    /Apache
  /AIXserver
    /Scripts
```

In this example, we would be mounting a total of three directories, from two systems.

To mount a directory, you use the IBM i command MOUNT. Here is an example:

```

                                Add Mounted FS (MOUNT)

Type choices, press Enter.

Type of file system . . . . . > *NFS          *NFS, *UDFS, *NETWARE
File system to mount . . . . . > 'linuxsys:/opt/WebSphere/AppServer/installed
Apps'
Directory to mount over . . . . > '/NFS/linuxsys/WebSphere'

Mount options . . . . . 'rw,suid,retry=5,rsz=8096,wsz=8096,timeo
=20,retrans=5,acregmin=30,acregmax=60,acdirmin=30,acdirmax=60,hard'

Code page:
  Data file code page . . . . . *BINARY      1-32767, *ASCII, *JOBCCSID...
  Path name code page . . . . . *ASCII       1-32767, *ASCII, *JOBCCSID

                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

The parameters of interest are:

Type of file system

Specify *NFS.

File system to mount

This parameter uses the format `hostname:filesystem`. So, in our example, we are mounting the file system we exported, which we specified as `opt/WebSphere/AppServer/installedApps` from the system named `linuxserver.mycompany.com`.

You can use either an IP address or a host name for the `hostname` portion of this parameter, but the `filesystem` portion must match what you typed in your export file.

Directory to mount over

The next parameter is the local mount point. This is the IFS directory we created above for mounting.

Mount options

The values shown above are the defaults, and we recommend that you do not change them. The two key values relating to TURNOVER® for iSeries are the first two shown. The first one, `rw`, indicates that the file system should be mounted as read/write. The second one, `suid`, indicates that it is OK to set UID's. (The UID is how UNIX identifies the file owner.) TURNOVER® for iSeries will run APIs to do this for you, so NFS needs to know this.

Once you mount the file system, its contents are available in the IBM i IFS as if it resided natively on the IBM i system, and you can use the `WRKLNK` command to access it as follows:

```
WRKLNK OBJ('/NFS/linuxserver/WebSphere/*')
```

This command would reveal the contents of the `/opt/WebSphere/AppServer/installedApps` folder on the Linux system.

Step 4: Configure TURNOVER® for iSeries global defaults

Configure your TURNOVER® for iSeries just as you would for IFS support, with two exceptions. (Refer to the *Managing IFS Objects using TURNOVER® for iSeries* guide for more details.)

Both exceptions involve IFS authority.

The first exception involves a TURNOVER® for iSeries Global Default setting. Because NFS systems need QSECOFR to be specified as the root user in order to set file ownership and permissions, and because the existing QNTC can use a different user as long as Windows permissions are set up correctly, a new TURNOVER® for iSeries global default has been added.

APPENDIX E: PROMOTING FILES IN A UNIX FILE SYSTEM

The new global default is called **QNTC Adopted authority user**:

```

Set Global Defaults (SETGLOBE)

Type choices, press Enter.

Allow group checkout . . . . . > N          *SAME, Y, N
Use Delete Test Object rule . . > Y          *SAME, Y, N
Company name . . . . . > 'YOUR COMPANY, INC.'
Dft pre/post run cmd lib list . > *CURRENT  *SAME, *CURRENT, *APPL
Specify method of distribution . > *DISTRIB  *SAME, *FORM, *DISTRIB
IFS Adopted authority user . . . > TURNIFS   Name, *SAME, *CURRENT
QNTC Adopted authority user . . > TURNIFS   Name, *SAME, *CURRENT
IFS Archive subdirectory . . . . > DEFAULT   Character value
Source archive file . . . . . > ARCHIVESRC  Name, *SAME
Library name . . . . . > TURNARC          Name
Automatic form lock checking . . > *DTAOBJ   *SAME, *NONE, *FILE, *DTAOBJ

(Main Menu option 8, then option 5 – second page of defaults.)

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More
    
```

For an NFS file system, set the **IFS Adopted authority user** to QSECOFR instead of the default TURNIFS. This allows TURNOVER® for iSeries to connect to the NFS file system as the root user (QSECOFR on the IBM i) so it can set file ownerships and permissions.

(If your application also manages objects in the QNTC file system, then you would set the new **QNTC Adopted authority user** value to the user TURNOVER® for iSeries should use when setting Windows file permissions and ownership (the default value is TURNIFS).

Promoting from UNIX to Windows or Windows to UNIX

TURNOVER® for iSeries copies or moves items directly, and there can only be one active user. Therefore, if your application will promote from UNIX to Windows or vice versa, then you must create a user called QSECOFR in your Windows domain.

Step 5: Create a TURNOVER® for iSeries authority list for use by NFS systems

Since the beginning of TURNOVER® for iSeries IFS support, you have created a TURNOVER® for iSeries authority list to designate the authority settings TURNOVER® for iSeries should use when promoting objects in the IFS. For Windows support, you don't use the TURNOVER® for iSeries authority list, because Windows authorities are set up on the Windows server.

For NFS file system support, we have enhanced the TURNOVER® for iSeries authority list to permit the explicit list types of *STD and *UNIX. The *UNIX type allows you to define UNIX-style permissions for the objects; the *STD type list is used for objects that reside in the native IBM i root file system.

The definition looks like this:

```

8/15/18          Change IFS Authority List Object          YOURSYS
11:38:03

Type choices and press Enter.

Authority list object . . . . . NFSTEST
Description . . . . . NFS Test
Type . . . . . *UNIX          *STD, *UNIX
Owner (UID) . . . . . 0        0-4294967294
Group (GID) . . . . . 0        0-4294967294

Permissions:
User/Owner . . . . . *RW          *NONE, *RWX, *RX, *RW, ...
Group . . . . . *RW          *NONE, *RWX, *RX, *RW, ...
Other/Public/World . . . . . *RW          *NONE, *RWX, *RX, *RW, ...

F3=Exit  F12=Cancel

```

As you can see, you can indicate the owner (UID) and group (GID) that you want to set on the objects you promote, as well as the permissions for the User, Group and Other.

TURNOVER® for iSeries sets these values using the `chown` and `chmod` APIs.

Create a *UNIX authority list for as many levels of your application as is needed (for example, you might have one list for development, one for QA, and one for production).

NFS owner GID and UID values and IBM i profiles

TURNOVER® for iSeries forms that manage UNIX objects will RECOVER when replacing objects (action code R), if there exists on the IBM i no user profile with UID and GID values that match the NFS object owner on the NFS system.

We recommend that you create IBM i profiles for all NFS object owners, and set the UID and GID values for those profiles to match their NFS counterparts.

To manage these values for an IBM i user profile, use the **User ID number** and **Group ID number** parameters of the CRTUSRPRF, CHGUSRPRF or DSPUSRPRF IBM i commands. (Page down about 5 times on the command prompt to see these fields.)

SUMMARY

You should now be able to set up and test a TURNOVER® for iSeries configuration meant to manage objects on a UNIX server.

ADDITIONAL RESOURCES

If you have questions, comments, or concerns, please contact your local Technical Support representative as outlined under the [SoftLanding website](#).

The following URL leads to an IBM Redbook that is useful for understanding and configuring NFS on the IBM i.

<http://www.redbooks.ibm.com/abstracts/sg246551.html?Open>

INDEX

—*—

*PUBLIC authority 3

—A—

add line exit 33
 Adopted authority user global default..... 5
 APIs *see* commands
 application definition, IFS requirements
 Line defaults 2, 9
 Path defaults 2, 9
 type codes 2
 application type codes 13
 archive history, IFS 25
 archive subdirectory
 global default..... 5, 6
 object info 25
 archived IFS objects, managing 26
 authority for *PUBLIC 3
 authority list, 10
 authority list, TURNOVER® for iSeries 3, 10
 for explode paths 9
 authority, IFS
 about 3
 authorization list, IFS 3, 10, 12

—B—

building IFS forms 19

—C—

checked out IFS objects, list of 28
 checkout history 27
 commands
 automatic default linked 37
 TADDIFSCMD (Add IFS line command) 15, 18, 21
 TADDIFSCMI (Add IFS line command - Interactive) 15, 21
 TADDIFSLIN (Add IFS object to form) .. 15, 18, 19, 20, 21
 TADDIFSLNI (Add IFS object to form – Interactive) 15, 19, 20
 TCHKINIFS (Check in IFS object) 15, 17
 TCHKOUTIFS (Check out IFS object) 15, 16
 configuring TURNOVER® for iSeries for IFS 5
 cross-reference info, IFS 32

—D—

defaults for IFS, global 5
 distributing IFS objects 23
 distribution history 27
 distribution subdirectories 23

—E—

exit points

add line exit 33, 34
 manual check-in exit 32
 post add line exit 34
 post checkout exit 32
 pre-checkout exit 33
 explode paths for authority lists for 9

—F—

file system
 Windows 41
 File system
 UNIX 45
 files, IFS
 TIFSARCF (IFS Archive Index file) 31
 TIFSAUTF (IFS Authority List Object file) 31
 TIFSCMDf (IFS Linked Commands file) 31
 TIFSCONF (IFS Conflict file) 31
 TIFSDFTF (IFS Application Defaults file) 31
 TIFSEXPf (IFS Application Explode List file) 31
 TIFSEXTf (Path Extension file) 31
 TIFSLINF (IFS Form Line file) 31
 TIFSOFFf (IFS Offline Archive Index file) 31
 TIFSOUTf (IFS Checkout file) 31
 TIFSTYPf (IFS Application Type Code file) 31
 finding TURNOVER® for iSeries information about IFS
 objects 25
 Form Maintenance functions for IFS 18
 forms, IFS
 building 19
 running 22

—G—

global defaults for IFS 5
 global type codes for IFS 1, 6
 group, primary 3, 12

—H—

how to manage IFS object 15

—I—

IFS
 application
 definition requirements 8
 Line defaults 9
 Path defaults 9
 archive history 2
 archive subdirectories 25
 authority, about 3
 authorization list 3, 10, 12
 checkout history 2, 25, 29
 distributing objects 23
 distribution history 2, 29

INDEX

distribution subdirectories	23	overview of IFS support	1
exit points		—P—	
add line	33, 34	Path defaults	2, 9
manual check-in	32	post add line exit	34
post checkout	32	post checkout exit	32
post-add line	34	pre-checkout exit	33
pre-checkout	33	primary group	3, 12
global defaults	5	problems	<i>see troubleshooting</i>
IFS Adopted authority user	5	promotion method	9
IFS archive subdirectory	5	—R—	
object authority	3	running IFS forms	22
objects, managing	<i>see managing IFS objects</i>	—S—	
overview	1	setting up TURNOVER® for iSeries for IFS	5
setting up for	5	subdirectories	
troubleshooting	35	archive	25
type codes		distribution	23
application	13	summaries	
global	1, 6	Form Maintenance functions for IFS	18
IFS 2	<i>see post checkout exit</i>	IFS files	31
IFS 3	<i>see manual check-in exit</i>	TURNOVER® for iSeries commands for IFS	15
IFS 4	<i>see add line exit</i>	—T—	
IFS 5	<i>see pre-checkout exit</i>	TADDIFSCMD (Add IFS line command)	15, 18, 21
IFS 7	<i>see post add line exit</i>	TADDIFSCMI (Add IFS line command - Interactive)	15, 21
—J—		TADDIFSLIN (Add IFS object to form)	15, 18, 19, 20, 21
java programs, optimizing	37	TADDIFSLNI (Add IFS object to form - Interactive)	15, 19, 20
—L—		TCHKINIFS (Check in IFS object)	15, 17
Line defaults	2	TCHKOUTIFS (Check out IFS object)	15, 16
authority list	9	troubleshooting	35
—M—		Windows support	42
managing archived IFS objects	26	TURNOVER® for iSeries authority list	3, 10
managing IFS objects		type codes	
adding them to forms	15, 18, 19, 20, 21	application	13
checking them in	15, 17	global	1, 6
checking them out	15, 16	—U—	
linking commands to IFS form lines	15, 18, 21	UNIX	
manual check-in exit	32	configuring	46
method, promotion	9	prerequisites	45
—O—		supported systems	45
object authority	3	using exit points to perform cross-referencing	32
object history, IFS	25	—W—	
object info sources		Windows support	
checked out objects	19	configuring	42
IFS archive history	2	prerequisites	41
IFS distribution history	2, 29	requirements for	41
IFS object history	2, 25, 29	troubleshooting	42
objects, distributing	23		
objects, managing	<i>see managing IFS objects</i>		

