SEE/Change

Supplementary Manual

(Incorporating PE's 4.2001 to 4.5003)



www.thenon.com

Many of the world leading companies use Thenon's products to change manage and test their software.

Thenon – designers of SEE/Change, the leading iSeries change management product.

Contents 1

1	CONTENTS	2
2	INTRODUCTION	9
3	SUMMARY OF PE'S	10
4	GENERAL13	
4.1	Fast Track options to prime functions (4.4301)	13
5	CONFIGURATION MANAGER	14
5.1	Object Authorities (4.3000)	14
5.2	Work With Object Authority (4.3004)	14
5.3	TAL Message Handling (4.3004)	15
5.4	CR Security: @NDO (4.3004)	15
5.5	ILE Set Up (4.3004)	16
5.6	ILE Compilation commands (4.3006)	17
5.7	Library Management (4.3006)	17
5.8	Changes to OCL36 / *INTERPRE2 (4.4000)	19
5.9	Cross referencing external libraries (4.4002)	20
5.10 5.10. 5.10. 5.10. 5.10. 5.10.	 Archive & CR library management (4.4003) Implementing the new archiving feature Using the archive feature Archive level management Compatibility with WRKLIBSTS (Work with library status) 	20 21 21 21 21 22
5.11 5.11. 5.11. 5.11. 5.11.	Override Processing Enhancements (4.4003)1Introduction2Configuration3Important points in setting up overrides for ILE parts4Overuse of overrides	22 22 22 24 24
5.12	Supplemental group profiles (4.4003)	25
5.13 5.13. 5.13.	Updating Module with changed objects only (4.4004) Introduction Configuration	25 25 25
5.14 5.14. 5.14. 5.14.	Extending the development lifecycle (4.4004)1Introduction2Configuration of the DevProd system3Special set up considerations	27 27 27 28
5.15	Source Pools Limit Increased (4.4004)	29
5.16	Support for File Transfer Protocol (FTP) (4.4100)	29
Issue 7	2: November 2005	Page 2 of 262

5.16.1	Configuring a system to use FTP	29		
5.16.2	Configuration Summary	31		
5.17	Overrides (4.4200)	31		
5.17.1	Object Type Override for a CR Type	31		
5.17.2	Object Overrides	33		
5.17.3	Support for production systems pre - 4.4200	34		
5.18	Authorise Release Distribution (4.4200)			
5.19	Automatic Release Handling (4.4200)	36		
5.20	Overrides for DMS submit RCVRLS (4.4201)	37		
5.20.1	Introduction	37		
5.20.2	Configuring the Overrides for DMS submit RCVRLS	38		
5.21	System Configuration (4.4202)	39		
5.21.1	Long Password Support	39		
5.22	Authorisation (4.4202)	40		
5.22.1	Authorisation for Users to Redevelop a CR	40		
5.23	SQL Support (4.4300)	40		
5.23.1	Introduction	40		
5.23.2	Configuration	41		
5.24	Bulk Management of Database Overrides (WRKAPPOVR) (4.4301)	44		
5.25	Suffix for Restore of DMS library (4.4301)	47		
5.25.1	Introduction	47		
5.25.2	Configuration	47		
5.26 .1	Archiving and Work Library Management (4.4301)	49		
5.26.2	Introduction	49		
5.26.3	Work with Library Status	49		
5.26.4	Consolidated archive library	49		
5.26.5	Pros and Cons	49		
5.26.6	Amalgamated system	50		
5.26.6	Work with Library Status (WRKLIBSTS) display	50		
5.26.7	SEE/Change Automatic clean up (SEEAUTO) command	51		
5.26.8	Recommended implementation	51		
5.27	Support for non-SQL trigger programs (4.4301)	51		
5.27.1	Configuration	53		
5.28	Take on Jobs Console (4.4304)	53		
5.29	Update Source Register (UPDSRCREG) (4.4304)	56		
5.30	Update Non Source Based Objects Register (UPDNSOREG) (4.4304)	57		
5.30.1	Message File Handling – Pre 4.4304	59		
5.30.2	Message File Handling – Post 4.4304	60		
5.30.3	Re-registering a message file.	61		
5.30.4	Further message file notes	62		
5.31	Cross Referencing take on aid (4.4304)	62		
5.32	REFXREF enhancements (4.4304)	63		
5.32.1	New parameter to reuse cached compilation data option	64		
5.32.2	Audit report	64		
5.33	Functions Added to User Authorisation Panel (4.4304)	64		
5.34	Enhanced Electronic Movement Authorisations (4.4304)	65		
5.34.1	Establishing Electronic Authorisation	65		
5.34.2	Granting and Revoking Movement Authorisations	66		

5.34.3 5.34.4 5.34.5 5.34.6	Working with Movement Authorisations Unauthorised movements Electronic Movement Authorisation by Message Migration from earlier releases.	67 67 68 68
5.35	Updating Object Source Details for Delivered Objects (4.4304)	69
5.36 5.36.1 5.36.2 5.36.3	Release Forwarding (4.4305) Introdiction Configuring Release Forwarding Purging the Release Packet Archive	70 70 70 76
5.37	Enhanced ILE Remote Delivery Method (4.5002)	77
5.38	Create Object Type (CRTOBJTYP) Command (4.5002)	78
5.39	Site Specific Configuration having no Database Library (4.5002)	79
5.40	Cross Referencing used with User Defined Object Types (4.5002)	80
5.41 .1 5.41.2 5.41.3 5.41.3 5.41.4 5.41.5	Stream File Support (4.5003) Introduction Overview Folders Folder Registration Further Information on Stream File Processing	81 81 82 83 84 93
5.42	Document Library Object (DLO) Support Withdrawal (4.5003)	101
6 F	ROBLEM MANAGER	102
6.1	Investigation request panel changes (4.3003)	102
6.2.1 6.2.2 6.2.3	Replacement for Office Vision (4.4202) Introduction Work with OMS text Documents (WRKOMSDOC) Sample Documentation Exit Program (O#DEXT)	103 103 103 105
7 C	HANGE MANAGER	106
7.1	Database Triggers (4.3000)	106
7.2 7.2.1 7.2.2 7.2.3	Cross Referencing (4.3006) Switching on Initial cross reference data population What's new to see?	106 106 107 107
7.3	Library creation exit program (4.3006)	112
7.4	Year 2000 Compliance (4.4000)	113
7.5	New exception report for cross referencing (4.4001)	113
7.6	New user exit program (4.4001)	113
7.7 7.7.1 7.7.2 7.7.3	Changes to ILE processing (4.4002) Changes to the OMS440 report Changes to application configuration Changes to application set up	113 114 114 114
7.8	Bound programs meeting in an environment (4.4002)	114
7.9	Changes to controlled deletions (4.4002)	115

7.10	Check CR Validation to *TST or *RLS (4.4003)	115
7.11 7.11.1 7.11.2	SQL Support (4.4300) Promoting / Reverting SQL-based objects Further SQL related topics	115 115 118
7.12 7.12.1 7.12.2	Database Trigger Support (4.4301) Introduction Preserving existing trigger definitions	119 119 119
7.13 .1 7.13.2 7.13.3 7.13.4 7.13.5	Cross Referencing (4.4301) Introduction Object Browser Release Packaging Additional movement controls Restrictions	120 120 121 122 122 122
7.14 7.14.1 7.14.2 7.14.3	Movement reporting & diagnosis improvements (4.4301) Introduction LSTMVTLOG (List Movement Logs) New Change Manage fast track options	123 123 123 124
7.15	Improved control of Reversion movements (4.4301)	124
7.16 7.16.1 7.16.2	Allow reset of Ready for Release status (4.4301) Introduction Configuration	125 125 125
7.17 7.17.1	Stream File Support (4.5003) The Movement Process	127 127
8 D	EVELOPMENT MANAGER	128
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism	128 128 129 130 131 132 132 133 134 134 135 135 135 135 136 136
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000)	128 128 129 130 131 132 132 133 134 134 134 135 135 135 135 136 136 137
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2 8.3	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000)	128 128 129 130 131 132 132 133 134 134 135 135 135 135 135 136 136 137 137
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2 8.3 8.4	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000) IMPORT/IMPORTLIB Upgrade (4.3001) Physical file Triggers (4.3002)	128 128 129 130 131 132 132 133 134 134 135 135 135 135 136 136 136 137 139
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2 8.3 8.4 8.5	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000) IMPORT/IMPORTLIB Upgrade (4.3001) Physical file Triggers (4.3002) Delete application part (4.3003)	128 128 129 130 131 132 133 134 134 135 135 135 136 136 136 137 139 139
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2 8.3 8.4 8.5 8.6	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000) IMPORT/IMPORTLIB Upgrade (4.3001) Physical file Triggers (4.3002) Delete application part (4.3003) CR Checking (CHKCR) (4.3003)	128 128 129 130 131 132 132 133 134 134 135 135 135 135 136 136 137 139 139 139 139
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2 8.3 8.4 8.5 8.6 8.7	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000) IMPORT/IMPORTLIB Upgrade (4.3001) Physical file Triggers (4.3002) Delete application part (4.3003) CR Checking (CHKCR) (4.3003) Support for ILE SQL (4.3004)	128 128 129 130 131 132 132 133 134 134 134 135 135 135 136 136 136 137 139 139 139 139 140
8.1 8.1.1 8.1.2 8.1.3 8.1.4 8.1.5 8.1.6 8.1.7 8.1.8 8.1.9 8.1.10 8.1.11 8.1.12 8.1.13 8.1.14 8.2 8.3 8.4 8.5 8.6 8.7 8.8	Changes to ILE processing (4.3000) Application Configuration Changes to Work with Parts using SCDM Creating a new ILE/Service program Updating an ILE/Service program Restructuring an existing ILE/Service program Deleting/Transferring ILE CR Parts The ILE Relationships Concurrent Development Issues Bind Capability Site Specific Modules (4.4300) Version Numbering for ILE Programs Development Limitations CHKCR - CR Checking The Movement Mechanism Programmer Defined Options (4.3000) IMPORT/IMPORTLIB Upgrade (4.3001) Physical file Triggers (4.3002) Delete application part (4.3003) CR Checking (CHKCR) (4.3003) Support for ILE SQL (4.3004) Work with Parts using SCDM (4.3004)	128 128 129 130 131 132 132 133 134 134 134 135 135 135 136 136 136 137 139 139 139 139 139 140 140

8.10	Programmer Defaults enhancement (4.3006)	143			
8.11	Data Queue Support (4.3006)				
8.12	Browse XREF Data (BRWXRFDTA) enhancement (4.4001)	143			
8.13	Bound and service program retrieval (4.4002)	143			
8.14	Hybrid ILE programs (4.4003)	144			
8.15	Non source based item search (4.4003)	144			
8.16	Database Management Commands (4.4004)	145			
8.17	Compile directives in SDA and RLU (4.4004)	147			
8.18	ILE Processing (4.4201)	147			
8.19	Display CR current environment (4.4202)	148			
8.20 8.20.1 8.20.2	Display Exported Procedures (DSPEXPPRC) (4.4202) Introduction Display Exported Procedures (DSPEXPPRC)	148 148 149			
8.21 8.21.1 8.21.2 8.21.3 8.21.4	Import process (IMPORTLIB) (4.4202) Introduction Import External Library (IMPORTLIB) Work with Import Register (WRKIMPREG) Import Object/Source Register (IMPORT)	149 149 150 150 151			
8.22 8.22.1	SQL Support (4.4300) Handling SQL source in a CR	152 152			
8.23	Improved source member retrieval display (4.4301)	156			
8.24 8.24.1 8.24.2	Overriding duplication of logical file members (4.4301) Introduction Configuration	157 157 157			
8.25	New object type DDMF (4.4301)	158			
8.26	Execute CR List (EXCCRLST) Utility Command (4.4304)	158			
8.27	Update Object Source File / Library (UPDSRCFLIB) (4.4304)	159			
8.28	SEE/Change Plug-in for WDSc (4.5000)	161			
8.29 8.29.1 8.29.2 8.29.3 8.29.4 8.29.5 8.29.6	File Attributes (4.5002) Introduction Configuration Overrides Attribute Retention in the CM Lifecycle Apply File Attributes (APYFATR) command. Consistency of configuration considerations.	161 161 162 162 163 165 165			
8.30	Supported File Attributes (4.5002)	167			
8.31	Ability to view Archived Source Members (4.5002)	167			
8.32 8.32.1 8.32.2 8.32.3 8.32.4	Stream File Support (4.5003) Folder Retrieval The Development Process Ending the Development Process Further Information on Stream File Processing (4.5003)	168 168 178 189 191			

9 R	ELEASE MANAGER	201
9.1	Restoring release packet names (4.4000)	201
9.2 9.2.1	Overrides for DMS submit RCVRLS (4.4201) Assigning an Override Id to a Release	201 201
9.3.1 9.3.2 9.3.3 9.3.4	Release Forwarding (4.4305) Sending a Release from the Development Centre Forwarding a Release from an Intermediate System Forwarding a Release from the command line Restrictions	202 202 202 203 204
10 A	UDIT MANAGER	207
10.1	Transfer history (4.4002)	207
11 c	OMMUNICATIONS MANAGER	208
11.1	Pack size limit (4.3003)	208
12 II	NTERFACES	209
12.1	Hawkeye interface (4.3006)	209
12.2	Import register upgrade for CASE tools (4.3006)	209
12.3 12.3.1 12.3.2 12.3.3 12.3.4 12.3.5	JD Edwards World software Interface (4.4003) Overview Configuring SEE/Change for JD Edwards Using the SEE/Change - JDE Interface JDE Library Overrides Data Area VTTL@	210 210 211 212 213 213
12.4 .1 12.4.2 12.4.3 12.4.4 12.4.5 12.4.6	The new ASSET Interface (4.4200) Introduction System Configuration Application configuration Recommendations for Best Practice ASSET object types Using the Interface	214 214 215 218 218 219
12.5	COOL:2E - V7.0 ILE compatibility (4.4201)	221
12.6	LANSA variables (4.4004)	221
12.7	High Availability Compatbility / EXCCRLST (4.4301) (4.4301)	222
13 s	UPPLEMENTAL PE NOTE INFORMATION	224
13.1	General Parameter Changes	224
13.2	Object type support	227
13.3 13.3.1	Software Performance Reports HDM 5034: ILE Program commands	228 247
13.4 13.4.1 13.4.2 13.4.3	Installation 4.2001 (CISC) 4.2002 (CISC) 4.3000 (RISC)	247 247 247 247 247

13.4.4	4.3001 (RISC)	247
13.4.5	4.3002 (RISC)	248
13.4.6	4.3003 (RISC) & 4.2003 (CISC)	248
13.4.7	4.3004 (RISC) & 4.2004 (CISC)	248
13.4.8	4.3005 (RISC) & 4.2005 (CISC)	248
13.4.9	4.3006 (RISC)	248
13.4.10	0 4.4000	248
13.4.11	4.4001	249
13.4.12	2 4.4002	249
13.4.13	3 4.4003	250
13.4.14	4.4004	251
13.4.15	5 4.4005	251
13.4.16	6 4.4100	251
13.4.17	4.4101	251
13.4.18	3 4.4102	251
13.4.19	9 4.4103	251
13.4.20	0 4.4200	252
13.4.21	4.4201	252
13.4.22	2 4.4202	252
13.4.23	3 4.4300	252
13.4.24	4.4301	252
13.4.25	5 4.4302	252
13.4.26	6 4.4303	253
13.4.27	4.4304	253
13.4.28	3 4.4305	253
13.4.29	4.5000	253
13.4.30	4.5001	253
13.4.31	4.5002	253
13.4.32	2 4.5003	254
13.5	SEE/Change Compatibility Charts	255
1351	WDSc plug-in compatibility chart	255
13.5.2	Compatibility 4 4000 through to 4 4301	256
13.5.3	Compatibility 4 4100 through to 4 5000	257
13.5.4	Compatibility 4.4200 through to 4.5003	258
13.6	CISC / RISC Considerations	259
407		
13.1	PE Delivery Change notification	260
13.7.1	Changes to SEE/Change PE delivery (4.4100).	260
13.7.2	How this affects you	260
13.8	User Exit Programs	260

14 INDEX 262

2 Introduction

This document is a supplementary amalgamation of all PE notes cut since version 4.2000 of the SEE/Change manual.

The document is organised by Manager and within this by PE.

3 Summary of PE's

This section contains a brief summary of the PE's that are incorporated into this document.

4.2001	HDM's Only
4.2002	HDM's Only
4.3000	RISC entry level ILE Implementation Programmer defined options Database Triggers Object authorities
4.3001	IMPORT / IMPORTLIB upgrade
4.3002	HDM's Only
4.3003 and 4.2003	Delete application parts CR Checking Pack size limit Investigation request panel changes New object types OVL, PAGSEG, PNL-NOSRC, SBSD, SRT, TCPIP
4.3004 and 4.2004	Source conversions WRKOBJAUT enhancements TAL message handling CR Security ILE Module Compilation strings Support for ILE SQL
4.3005 and 4.2005	HDM's Only
4.3006	Cross Referencing Library Management (WRKLIBSTS) Data Queue Support Library creation exit program SCDM protect text option Hawkeye interface Import register update ILE compilation commands
4.4000	Year 2000 compliance Restoring release packet names
4.4001	BRWXRFDTA command enhancement New exception report for cross referencing New RCVRLS exit program
4.4002	ILE enhancement ASSILEPGM Support for RPGLEREF Cross reference external libraries Bound programs meeting in an environment Bound and service program retrieval Changes to controlled deletion Transfer history
4.4003	Enhancements to Override Processing Supplemental Group Profiles Hybrid ILE programs CR Validation Check CR to *TST or *RLS Archive & CR library management JD Edwards interface

4.4004	Develop in Module Dev/Prod system – extending the development life cycle Database management commands Compile derivatives in SEU and RLU Source pools limit increased LANSA variables
4.4005	OS/400 V4R4
4.4100	Changes to PE numbering from hereon Support for FTP distribution
4.4101	HDM's Only
4.4102	HDM's Only
4.4103	HDM's Only
4.4200	IFS Support 4.4003 Override changes anomalies addressed Authorise release distribution Automatic release handling
4.4201	Overrides for DMS submit RCVRLS ILE Processing rework Support for prod systems pre-4.4200
4.4202	OS/400 V5R2 (Long Passwords / Long Library Lists) IMPORTLIB Replacement for Office Vision Documents Display Exported Procedures Display CR environment Authorisation for users to Redevelop a CR Remove source pool function
4.4300	Support for SQL object types (DDL / DML)
4.4301	Full support for Group and Site level Cross Referencing Improved Source Member Retrieval displays IFS File types limit of 4 characters expanded Bulk management of database overrides Archiving and Work Library Management enhancements Full support for Database Triggers and Trigger Program realignment Interface for use with High Availability products Improved reporting and diagnosis ability Improved reversion movement control Reset of Ready Release flag
4.4302	HDM's Only
4.4303	HDM's Only
4.4304	Electronic Authorisation Lists enhanced to cover redevelopment movements Non-Source Based Objects Registration Cross Referencing Take on Aid
4.4305	Release Forwarding Support for intermediate IFS environments
4.5000	WDSc plug-in phase 1 (Library objects)
4.5001	HDM's Only
4.5002	File attribute retention in development lifecycle Supported file attributes Enhanced ILE program delivery mode (Programs only) View archived source from archive library save file CRTOBJTYP command Site specific configuration having no database library Cross Referencing support for user defined object types

4.5003	WDSc plug-in phase 2 (Stream file support)	
--------	--	--

4 General

4.1 Fast Track options to prime functions (4.4301)

SEE/Change's can be tailored to suit large and small organisations alike. In PE 4.4301 the emergence of development staff that may also be responsible for other common CM functions, such as configuration or software deployment, has been further recognised by providing a series of *Fast Track* options on the main SEE/Change menu, thus saving navigation through the individual manager menus.

SEE	SEE/Change - iSeries (Develo	Change Management	System:	THNSPT
Fast track SEE/Ch	ange options: 21. Work with Change 22. Work with Change 23. Work with Release 24. Work with Transfe 25. Work with Transfe 26. Display Object Hi 27. Work with Applica	Requests Request Development s r Status - Outgoing r Status - Incoming story tion Configuration		
SEE/Change menus: 1. Problem Ma: 2. Change Man. 3. Developmen 4. Release Ma: Selection or comm	nager ager t Manager nager and	5. Communication 6. Communication 7. Audit Manager 8. Configuration	Manager - Manager - Manager	Software Network
F3=Exit F4=Prom F13=Information A	ot F9=Retrieve F12= ssistant F16=AS/400 ma	Cancel in menu		

The new options behave as follows:

Option	Function
21	WRKCHGRQS
22	WRKCRDEV
23	WRKRLS
24	WRKTFRSTS TYPE(*OUT)
25	WRKTFRSTS TYPE(*IN)
26	DSPOBJHST <prompted></prompted>
27	WRKAPPCFG

Options 1 through 8 retain their original functions.

5 Configuration Manager

5.1 Object Authorities (4.3000)

With version V3R1M0 of the operating system three new object authorities were introduced, these are object rights to alter and ref and the data right execute. These have been added to the object authorities panel.

5.2 Work With Object Authority (4.3004)

This program has been enhanced to allow better control of the subfile list displayed and include print facilities.

			SY1 Tes	t Development	: System		
			Work wit	h Object Auth	norities		
2=0 13=5	Change Show & ap	3=Cre	ate	4=Delete	5=Displa	y 6=	Print
Opt	Apply Lvl (P)	Object Name APY+	Library Name OMS*	SEE/Change Ref Id (P)	Authority Template	Process Template	
<u></u>	*OBJ *OBJ *OBJ	APYOBJAUT CRTRLSTAP	OMSOBJ OMSOBJ OMSOBJ	CLP CLP CLP	*YES *YES *YES		
	*OBJ *OBJ	ENDDMSJOB	OMSOBJ OMSOBJ	CLP	*YES *YES		
	*OBJ *OBJ	EXCOBJLST OMSLNS02	OMSOBJ OMSOBJ	CLP RPG	*YES *YES		
	*OBJ *OBJ	OMSTXT OMS216C	OMSDTA OMSOBJ	PF CLP	*YES *YES		
	*OBJ *OBJ *OBJ	OMS240C OMS241C OMS246C	OMSOBJ OMSOBJ OMSOBJ	CLP CLP CLP	*YES *YES *YES		
	020	01102100	0110020	0.11	100		More
F1=He	elp F3=H	Exit F4=Pr	ompt F8=Pr	int subfile	F9=Cmd F23	=More opt	F24=Msgs

The hierarchy of position to fields has been relaxed. Therefore there is no need to specify all the fields from left to right.

Two new operations have been included. a '+' and a '*' symbol. Specifying a '+' requests a position to like operation where a match is found with the preceeding characters then all subsequent records are displayed. In the above example APY+ finds a record starting with APY and then continues alphabetically. An '*' symbol requests a wild card operation where the characters preceding the '*' are selected then any subsequent characters are allowed. In the above example OMS* requests the program to show only libraries starting with OMS, only OMS is selected it will not then continue alphabetically. If you do not specify a '+' or an '*' then the subfile will select only records that have an exact match. Two new print facilities have been included. Option 6=Print against a subfile line will print details of the selected line. Pressing F8=Print List will produce a report of the entire contents of the list. If you use the new subsetting functions described above before pressing F8 = Print List the list printed will reflect the subsetted list.

The program has also been changed to operate in a *browse* mode only, allowing users to view or print details. A new entry has been set up in *Work with User Enrolment* to facilitate the authorisation of either update or browse only mode.

5.3 TAL Message Handling (4.3004)

Authorisation request messages have been enhanced to ensure that only the recipient of the message is allowed to answer the message.

If somebody answers an authorisation request on somebody elses message queue, the authority checking program will notice this and ignore the answer. The authority checker will also inform the respondant that they are not allowed to answer messages not addressed to them.

Note: Since the original message has been answered, it will be necessary to resend the authorisation request message.

5.4 CR Security: @NDO (4.3004)

A new function has been created to secure a CR library once it has left development. This is to prevent tampering with the contents of a CR between environments.

A new parameter @NDO (Non development owner) can be set up in parameter data to name the owning profile of CR libraries not in development. When a CR reaches Module/Integration the owner of and authority to the CR library is set to the value of @NDO.

Therefore setting @NDO to a value of QSECOFR will restrict the CR library to QSECOFR. If the CR is subsequently reverted to development, its original owner (Specified on the CR) is regranted rights to the library.

If @NDO has no value or a value of *SAME then no change of authorities is made.

Note: Change of authority takes place only on a physical CR movement and not when being checked to status *TEST or *READY.

ILE Module Compilation Strings (4.3004)

A new menu option has been added to menu SEECFG2 (Extended configuration menu), option 13 = Edit Module compilation commands. This is in addition to option 12 = Edit compilation commands.

5.5 ILE Set Up (4.3004)

A new command BLDILEREF (Build ILE References) has been created. This program populates the ILE parts of the SEE/Change database in a similar manner to UPDSRCREG (Update source register) which populates the OPM parts of the database.

This is a once only program that should run only in the following circumstances.

- You are new to SEE/Change and have ILE programs to initialise into the product.
- You have developed ILE programs outside of SEE/Change and would now like SEE/Change to take over control.

Before running the command you should have all your ILE objects in their live libraries. The module pool library should be configured in application configuration (WRKAPPCFG) and your modules should be within it.

Note: Site specific modules live in their respective site specific library.

The command will prompt you to enter the application, the library to scan (Each program library will have to be processed separately) and the source type and qualifier. There is no need to run the command over the declared module pool library as this is a function of the program (Unless your module pool library also contains programs).

Example configuration for base application and one site that has its own specific modifications library:

APPLICATION = FED	
-------------------	--

Base Application Libraries	Site LON Specific Libraries	Module Pool Library	
FEDPGMS	LONSPECIF	FEDPOOL	
FEDDATA	LONDATA		

BLDILEREF will need to be run twice:-

• Once for the base application program library;

BLDILEREF APP(FED) LIB(FEDPGMS) SRCT(*BAS) SRCQ()

This will also populate the module pool files for libraries FEDMODS and LONSPECIF.

And once for the site specific program library;

BLDILEREF APP(FED) LIB(LONSPECIF) SRCT(*SIT) SRCQ(LON)

Database libraries FEDDATA and LONDATA that contain only files are not required for processing.

A report is produced detailing the completion status of each object processed.

5.6 ILE Compilation commands (4.3006)

The commands CRTPGM and CRTSRVPGM that were hard coded and therefore not available for alteration have been softcoded. Messages OMX6003 and OMX6004 for CRTPGM and CRTSRVPGM respectively have been set up.

If you wish to change these messages that affect the defaults when creating new programs you should use the *Merge message file* (MRGMSGF) function to copy the message(s) from OMSMSGE to the user file OMSMSGU before manipulation.

5.7 Library Management (4.3006)

A new function has been created to help manage the various work libraries created automatically by SEE/Change. Basically it allows you to interrogate why a library exists and how long it has been resident on the system. You can select time periods for expiry of work libraries with a defined action to execute on expiry. For instance you could say, "Keep CR libraries for 90 days after the date that they go live, save them to tape then delete them." After the 90 days the library will be deemed expired.

This functionality comes in two main parts, the first is the enquiry panel where you can view and manipulate libraries, the second is a function that actually processes actions against libraries on expiry. This function could be plugged into a backup routine to automatically manage the work libraries.

Libraries that are saved to tape using this function are logged within SEE/Change, therefore if there is a future requirement for say an archive library , SEE/Change will prompt you to load the tape it knows the library was saved on to.

The enquiry panel is available on the extended configuration options menu, option *17=Work With Library Status*, or WRKLIBSTS from a command line.

An example of the library enquiry panel follows:

THENON C. Wo.	hange Management sy rk with libraries	stem.			
Resident Libraries Type options, press Enter. 1=Save 2=Change 4=Remove entry 5=Display					
Date	Date Date				
Opt Live Library Type	Saved Deleted	Status Comment			
0/00/00 O#00000102 CR	0/00/00 0/00/00	Active			
0/00/00 O#21003213 CR	0/00/00 0/00/00	Active			
0/00/00 O#21003214 CR	0/00/00 0/00/00	Active Phase 2 Li	b		
0/00/00 O#21003215 CR	0/00/00 0/00/00	Active			
0/00/00 O#21003302 CR	0/00/00 0/00/00	Active			
30/08/90 O#00000301 CR	0/00/00 0/00/00	Active			
5 2/12/93 B00000013 Archiv	e 0/00/00 0/00/00	Active			
16/02/94 B000000016 Archiv	e 1/01/91 0/00/00	Expired			
16/02/94 B000000017 Archiv	e 1/01/91 0/00/00	Expired			
16/02/94 B000000018 Archiv	e 0/00/00 0/00/00	Active			
More					
FISHER FISHER FISHER FISHER FISHER FISHER					
FID=CONTROL OPTIONS FIZ=Can	cei				

The panel shows each library in date order, you can position by 'go live' date. The default list is of resident libraries *pressing F10=Resident/Non Resident* will toggle the view to show libraries that have been removed from the system, giving you the oppurtunity to manually restore them.

Available options:

Option 1=Save

This option will allow you to manually save the library immediatly. On the view of non resident libraries option 1 allows the restore of the library.

Option 4=Remove Entry

This in effect deletes the line and should be used only to remove unwanted old records.

Option 2=Change or 5=Display

Will show the following panel.

: Library Details : aries T : : 1 : Library: B00000016 Status : Active :	
1 : Library : B00000016 Status : Active : :	
: Library resident : Yes : Op : : nt	
: Date CR went live: 16/02/94 Related CR. : 210031 / 11 : : Date saved : 0/00/00 : : Date Deleted : 0/00/00 :	
: Tape Volume: : : : : : : : : : : : : : : : : :	
: Expiry action: *DELETE (*NONE/*SAVE/*DELETE/*SAVDEL) : 5 : : : F3=Exit F12=Cancel :	
: : :	

This panel show the details of the archive library. It shows the library is still Active ie not expired. The panel also shows the number of the related CR that caused the archive to be created. If the library had been saved to tape the volume id of the tape would also be shown. There is also a free format comments line.

The expiry action for this library is set to *DELETE to delete the library. *SAVDEL implies save then delete.

You can set the default expiry actions etc by taking *F16=Control options* from the main subfile display. The following panel is shown:

THENON Change Management system.				
Library control				
CR Libraries				
Number of days to keep after CR goes live 10				
Action after expiry	ONE/*DELETE			
*Si	AVE/*SAVDEL)			
Archive Libraries				
Number of days to keep after CR goes live 10				
Action after expiry	ONE/*DELETE			
*S/	AVE/*SAVDEL)			
Scheduled cleanup				
Tape device to use during automatic cleanup TAP01				
Tape in drive message during automatic cleanup . *MESSAGE (*1	MESSAGE/*ASSUME)			
F3=Exit F9=Command F12=Cancel				

The number of days to keep a library after the CR goes live (On the development system) and the expiry action can be set independently for CR libraries and archive libraries.

The scheduled cleanup options refer to the automatic cleanup program mentioned earlier. Specify a tape device for any save operations and a message action. A message can be sent to the system operator prompting them to insert a tape for the save of SEE/Change libraries. Overnight this could be awkward so you can set the message option to assume the tape is in the drive already.

The automatic cleanup operation is executed via a new command SEEAUTO. There are no parameters in the command. You may place the command SEEAUTO in your own nightly routines if required.

5.8 Changes to OCL36 / *INTERPRE2 (4.4000)

Currently OCL code which is of source type *INTERPRET remains resident in each environment upon promotion. That is code is copied from environment to environment rather than being moved.

A number of customers have requested that this be changed so that OCL36 source members are actually moved rather than copied.

To facililate this we have created a new source usage of *INTERPRE2 to mark it as a special type of *INTERPRET which behaves slightly differently (ie moves).

OCL36 has been changed to be of source usage *INTERPRE2 by the installation program.

IMPORTANT: If you do not want to handle OCL36 in this way you must change its source usage back to *INTERPRET via command CHGOBJDFT (Change object defaults).

5.9 Cross referencing external libraries (4.4002)

A new file has been created to support the collection of cross-reference data from libraries not part of the configured application. This can be used in the situation where software from a third party is being used as a base for modifications. The third party library does not comprise part of the configured application but does complete the cross-reference. I.e. a bespoke logical over a physical in the third party base library.

Use the new command WRKXRFLIB (Work external reference libraries) to enter the names of the external libraries.

5.10 Archive & CR library management (4.4003)

It is now possible to specify a single archive library. This library will store the current archive libraries as save files within it. Also CR libraries upon promotion to live will be saved into a save file and stored into the single archive library.

The archive library is specified at application level.

The result of this change means that the only work libraries present on a system will be CR libraries that are not yet in the live environment. When a CR is reverted or a release is created which is packed from the CR library, the requisite archive/CR libraries are restored from the single archive library. Upon completion of the operation any relevant libraries are repacked into the archive library.

For example a CR is promoted to live and its CR library is saved to a save file and stored in the single archive library. If a release were sent that required objects from the CR library, the CR would be restored, the objects used then the CR would be resaved to the archive.

This enhancement will greatly benefit sites where large numbers of CR's are generated.

To activate this new feature simply key a library name into the new field for Archive library in the application details panel as shown below. To disable the feature the field should be blank.

THNS20 SY1 Test Development System	
Work with Application Details	
Application code : AP1	
Application description : Demo application 1	
Development centre system(P): SY1 SY1 Central DEV	
Development centre site(P): SI1 SI1-Dev for ap1	
Application ASP : 0 $0-16 (0 = \text{use CRTLIB cmd dft})$	
Application job description : AP1JOBD	
Application message file library : AP1SAV	
Default CASE tool (P): *NONE Model/Set Name :	
Cross-application documentation : N (N)one/(A)ppl/(D)ate	
Re-compile objs at remote sites ?: Y (Y)es/(N)o	
Re-compile objs at Dev Centre ? : N (Y)es/(N)o	
Distribute source code ? : Y (Y)es/(N)o	
Distribute object override info ?: Y (Y)es/(N)o	
Message file operations : D (M)erge/(D)uplicate	
Multiple versioning ? Y (Y)es/(N)o	
Planned concurrent development ? : Y (Y)es/(N)o	
Auto revert if promote errors ? : N (Y)es/(N)o/(P)rompt	
Number of archiving levels : 01 (0-99) Archive Library : APIARCL	
Configuration complexity : 3 1=Simple 2=Intermediate 3=Complex	
F1=Help F3=Exit F4=Prompt F9=Cmd F12=Cancel F16=Update	
F18=Extended Options F24=Messages	

5.10.1 Implementing the new archiving feature

In order to start using the new feature, all the existing archive and live CR libraries on your system need to be saved into the archive library.

Firstly enter a library name into your application configuration. The library must be an existing library.

Enter the command PCKALLARC (Pack all archives) press F4

Enter the system and application codes and ensure the *pack* field = 'P'

If you wish to disable the feature, unpack all the libraries by using the same command with the pack field set to 'U'. Then blank the library name in application configuration, which acts as an on/off switch for the new archiving method.

5.10.2 Using the archive feature

When the new archive feature is operating the only visible change will be the addition of two spool files to the output produced by the MOVCR job during promotion.

- **UNPACK** Messages from the unpacking program that unpacks libraries prior to a revert operation or a release movement.
- **PACK** Messages from the archive packaging program that packs libraries after a promote to live or release movement.

5.10.3 Archive level management

With the new feature enabled it is impractical for the promote process within SEE/Change to manage the number of archiving levels. Since the objects to be purged beyond the specified level exist in save files it could be a time consuming task to open the required save files and perform the purge during promotes. For this reason a new function has been created called PRGARCOBJ (Purge Archived Objects). This command can be executed on an as needed basis to clean up the no longer required objects. This would have the benefit of decreasing the size of the archive library. And therefore should be scheduled as part of your periodic cleanup activities.

The command PRGARCOBJ should be submitted in an environment with the correct environment libraries (i.e. OMSOBJ and OMSDTA).

5.10.4 Compatibility with WRKLIBSTS (Work with library status)

The new archiving method is supplied as an alternative to WRKLIBSTS (See PE notes 4.3006). The two methods should not be used together. Ensure the housekeeping job SEEAUTO (Released in 4.3006) is removed from your scheduled processes before implementing the new archiving method.

5.11 Override Processing Enhancements (4.4003)

5.11.1 Introduction

Previously SEE/Change's supported the use of just one override record per AS/400 object. Although the override could be applied at CR Type, Object type or specific object level. Only one alternative target library could be specified.

This restriction has been removed. It is now possible to target a given object at up to 99 different target libraries.

This gives increased flexibility for application configuration and may in some cases simplify configuration where SEE/Change 'sites' have had to be configured. If a site were configured simply to overcome a problem delivering to a library, the overhead of that site could be removed.

The limit of 99 target libraries is by environment. Therefore you could construct a configuration where an object is delivered to perhaps 10 Integration libraries but only 3 Acceptance libraries and finally on to any number of live libraries (up to 99).

5.11.2 Configuration

Outwardly the changes to SEE/Change seem slight. Only the three panels where overrides are configured are changed. They now include a field to accept the number of target libraries you would like to configure.

Take for example the following panel, which shows CR type overrides with the new field highlighted.

THNS20 SY1 Test Develor	ment System	n				
Work with Inplication Overrides - by CR Type						
WOIK WICH Application ove	.iiiucs by	y on Type				
Application : AP1 Demo application 1 System: SY1						
Object Group : Obj: Base application Number of target libraries per CR type:	lon l					
· · · ·	Overr	riding target l	ibraries			
CR Type	Live/Prod	Accept/0A	Intear/Tst			
Program Bug Fixing	1100/1100	1000000,011	OVBLIB1			
CP Closed - no development			OVIGIDI			
Emorgonou Fix						
Ameliantian Madifiantian						
Application Modification						
New development						
Procedure Error						
User Error						
Program modification due to x-references						
			Dottom			
			BOLLOIII			
FI=Help F3=Exit F9=Cmd F12=Cancel F13=	-Repeat F24	1=Messages				

In the panel shown above, the number of target libraries is set to one. When this number is changed, the subfile rebuilds to show the extra available libraries as shown on the next panel.

THNS20 SY1 Test Develop	ment System					
Work with Application Ove	rrides - by (CR Type				
Norn wron npprrodoron ovo	111400 21	011 1900				
Application : AP1 Demo applicati	on 1	System:	SY1			
Object Group : Obj: Base applicati	on					
Number of target libraries per CR type: 3						
	Overri	ding target 1	ibraries			
CR Type	Live/Prod	Accept/OA	Integr/Tst			
Program Bug Fixing		<u>-</u>	OVRLTB1			
Program Bug Fixing			OVRLTB2			
Program Bug Fixing OVRIDE						
CR Closed - no development						
CR Closed - no development						
CR Closed - no development						
Emergency Fix						
Emergency Fix						
Emergency Fix						
Application Modification						
Application Modification						
Application Modification						
••			More			
F1=Help F3=Exit F9=Cmd F12=Cancel F13=	Repeat F24=I	Messages				

With the 'number of target libraries' set to 3 the subfile list displays 3 lines for each of the CR types, allowing 3 override libraries to be configured.

The 'specific object' override panel caters for the multiple levels in a slightly different way:

SY1 Test Development System Work with Object Override Details Enter the movement override specifications for object: Application: AP1 Demo application 1 DSPF Level: *BAS Type/Attr: *MENU Object: FRED Number of target libraries per object: 2 Locate System/Site (P): System/Site SY1 SY1 Centra Current target library sequence : 01 (Dev System) Configuration Overrides Live/Production Lib : AP1SY1PL Acceptance/QA Lib : APISY1PA Mdl/Integration Lib : APISY1PM More... F1=Help F3=Exit F4=Prompt F9=Cmd F12=Cancel F23=Dlt overrides F24=Msgs

The field 'Number of target libraries per object' is the same as the CR type overrides panel. However because this panel has no subfile it displays each override record on its own panel. There is a new field 'Current target library sequence' to show the sequence number of the override you are currently viewing. Pressing *roll up* on this panel will increment that number until the 'Number of target libraries' has been reached. At this point the panel will move to the next configured system starting at override sequence number 1.

5.11.3 Important points in setting up overrides for ILE parts

Since ILE parts are configured in SEE/Change with two registry entries (i.e. RPGLE_SRC & RPGLE_MOD or RPGLE_SRC & RPGLE). The care must be taken when entering overrides.

If you do not have compile switched on (Compile during promote). Then SEE/Change effectively ignores the _SRC entries therefore any overrides need only be put onto the _MOD or program *object (plain RPGLE)*.

If compile is switched on then SEE/Change processes the _SRC records, in which case it is essential to put the overrides on _SRC entries in SCDM. To make the OMS440 report easier to read it is recommended that if compile is switched on then override entries are added to both the _SRC **AND** the _MOD or program *object*.

5.11.4 Overuse of overrides

Any object, program or file can be overridden to multiple target libraries. When overriding ILE programs formed out of modules, it is advisable to keep modules together in the same library as their dependent program counterparts. For example a configuration that distributes ILE programs to libraries A, B and C yet distributes their component modules to Y and Z can quickly become complex.

5.12 Supplemental group profiles (4.4003)

SEE/Change now recognises user profiles that are attached to supplemental group profiles. Therefore a given user is authorised to a function if the function is assigned to its group profile (as before), or assigned to one of its (OS maximum of 15) supplemental group profiles.

5.13 Updating Module with changed objects only (4.4004)

Otherwise known as Develop in Module.

5.13.1 Introduction

It is now possible to promote a CR to the MDL/Integration environment and still continue working within the CR. At any time after working in a CR that has been promoted to MDL/Integration, the CR can be repromoted, (subject to passing validation checks). During the repromotion, only the objects that have been changed in the CR since the last promotion will be moved to the MDL/Integration environment.

This means that a change to a required object no longer requires a full revert of the CR back to development, rather the object can be changed and repromoted ready to test again.

5.13.2 Configuration

To configure this option go to the *Work with Application Configuration* – *Libraries* panel as below and take new option *15* = *Environment settings*.

THNS20	Work	SY1 Test Develo with Application Conf	pment System iguration -	m Libraries	
Application : AP1 Demo application 1 Local Sys. : SY1					
12=Obj typ overrides 13=CR type overrides 15=Environment settings					
Opt System	Туре	-Type:Description	T	arget libra: Accept/OA	ry Ovr Integr/Tst
SY1 Centra	Dev	Obj: Base application Obj: Site SI1-Dev for DB : Base SI1-Dev for JDE: Site SI1-Dev for	AP1SY1PL AP1SY1DL0 LIB1	AP1SY1PA AP1SY1DA0 LIB2	APISYIPM APISYIMDL LIB3
SY2-Europe	DvPr	Obj: Base application DB : Base A test site JDE: Site A test site	Live/Prod	Accept/QA	
					More
F1=Help F3=Ex:	it F	5=Refresh F9=Cmd F12	=Cancel F1	6=Update Fi	24=Messages

This presents a new panel that allows configuration changes to be made at the environment level. The first flag to utilise this is the *Develop in Module/Int*. The flag is only available at the MDL/Integration level. Future flags will be available at single or multiple environment levels.

To activate this new facility simply type a Y for the flag under the MDL/Integration column as shown below.

Work with Appl:	GY1 Test Development Sy cation Configuration -	rstem • Environment flags
System : SY1	SY1 Central DEV	
Application : AP1	Demo application 1	Local Sys. : SY1
Description Develop in Module/Int	Live/De	Target Environment v Accept/QA Integr/Tst Y
		Bottom
F1=Help F3=Exit F5=Refree	sh F9=Cmd F12=Cancel	F24=Messages

It will now be possible to promote the CR to MDL/Integration and still work on its contents. When working on the contents any compilations are still targeted at the CR library, you never compile directly into the MDL/Integration environment. To refresh the MDL/Integration libraries, re-promote the CR and only the objects that have been changed will be promoted. Any cross-referencing recompilations, (as long as crossreferencing is enabled), will be performed for the object being promoted.

Objects in the CR that have changed since the last promotion to MDL/Integration are highlighted in one of two ways. By default the objects will be coloured (*pink*), but green screen users can change this to be *underline* instead. To change the colour/underline setting use F17=Pgmr options from the main Work with parts using SCDM panel and then F17=Change user defaults from the Programmer options window.

When a CR contains objects that have changed the CR status will be shown as *Chgd Items* rather than Mdl/Integration. When the CR is at status *Chgd Items* a promote to MDL/Integration is allowed, if there are no changed items in a CR at MDL/Integration you are not allowed to repromote the CR.

If you are using cross referencing to recompile objects (i.e. *FULL cross referencing in application configuration) you will not be allowed to change or edit any database related application parts once the CR has gone to MDL/Integration. To change a database part you should revert the CR first in the normal way.

If you are not fully utilising cross referencing, (i.e. *NONE or *DATA in the application configuration), you will be able to change database parts but it is up to you to either recompile affected program in your CR, or if you compile on promote you should flag the affected program(s) using option *31=Refresh MDL* in SCDM to force the program(s) to be repromoted and thus recompiled.

If you are using general parameter @NDO to restrict authority to CR's in conjunction with this feature, authority will no longer be revoked when a CR is promoted to MDL/Integration.

5.14 Extending the development lifecycle (4.4004)

5.14.1 Introduction

It is now possible to hold the status of source members to prevent retrieval into a CR until the object has been promoted live at a nominated, local, production system. This has the effect of creating another test environment on the development system (live).

The nominated production system that releases source locks when objects are promoted to it is called the Development production system (DevProd). SEE/Change application configuration panels have been altered to show the DevProd system next to the main development system.

When a system has been nominated as the DevProd system there is no longer the promotion step of RDY/Release, that is CRs may be directly promoted to live on the development system without having to go to status RDY/Release or being assigned to an actual release. To send a CR to the DevProd system it will be necessary to assign the CR to a release, this can be done after the CR has been promoted to live on the development system.

5.14.2 Configuration of the DevProd system

The following panel shows the library configuration screen with the DevProd system directly underneath.

```
      SY1 Test Development System
Work with Application Configuration - Libraries

      Application
      . . . . : AP1 Demo application 1
      Local Sys. : SY1

      12=0bj typ overrides
      13=CR type overrides
      15=Environment settings

      Opt System
      Type Type:Description
      ------ Target Library ------ Ovr
Live/Dev Accp/Dev Integr/Tst

      SY1 Centra Dev
      Obj: Base application AP1SY1PL
      AP1SY1PA AP1SY1PM
DB : Base S11-Dev for AP1SY1DL0
      AP1SY1DA0 AP1SY1MDL

      SY2-Europe
      DvPr
      Obj: Base application AP1SY2PL
DB : Base A test site AP1SY2DL
      Bottom
```

This feature can be enabled within SEE/Change via new setting in the application configuration. The following panel shows an example:



You must specify the system code for the DevProd system and also the site under which the OMS database for that system is configured. Before allowing any source retrieval into a CR on the development system, SEE/Change will look at the database on the DevProd system to ensure that the object in question has reached live (if *Unlock Source at Dev-Prod System* is Yes). If not, the programmer will not be allowed to retrieve the requested object except via concurrent development in the normal way.

An error message and prevention of the requested retrieval will result if SEE/Change is unable to resolve to the database library indicated by the DevProd system & site configured. To ensure that SEE/Change does correctly resolve to the DevProd system, you must ensure that the OMS application is correctly configured. To do this work with the application configuration, WRKAPPCFG, and then select function 'F21=Include Thenon app'. Take option '12=Where used' against the OMS application and press enter to display the 'Work with Application Configuration – Libraries screen'. Ensure that the OMS application shows the correct system libraries for your DevProd system. A DevProd system must be local i.e. on the same machine as the Development system.

5.14.3 Special set up considerations

For an established production system it is possible to simply activate the feature in the method described above. However, if you are creating a new production system, (i.e. new OMSDTA database), to work with an established development system, extra consideration is required.

This is because the new production system will have no history of software deliveries, therefore SEE/Change will always prevent retrieval into a CR since a record of delivery to the DevProd system can not be found. To circumvent this you can provide the DevProd system with a false history by copying file XCR across to the DevProd database. Select only records where CRSTAT = '09' to ensure only live records on the development system are copied. Nb. If your DevProd system is at a lower SEE/Change release than the Development system, the functionality will still work but you will have to copy file XCR with *MAP *DROP.

This development helps a number of SEE/Change customers, if you think this might benefit your system we recommend that you contact your usual Thenon support team for further advice before enabling the feature.

5.15 Source Pools Limit Increased (4.4004)

The maximum number of base source pools for an application has been increased from 25 to 99.

5.16 Support for File Transfer Protocol (FTP) (4.4100)

SEE/Change has been enhanced to allow all types of transfer to be accomplished via FTP over TCP/IP. The new support has been implemented seamlessly into the current transfer processes. Therefore, apart from some initial setup the actual communications method in use (SNADS or FTP) will not be apparent to the user.

5.16.1 Configuring a system to use FTP

5.16.1.1 Prerequisites

The FTP processing relies on the configuration of the OMS (SEE/Change) application, in that the development system must have the correct library configuration for SEE/Change at all production systems. This is so the FTP "send" function can send data to a known production SEE/Change library. FTP distribution identifies a system's data library as the FTP target library. This differs from SNADS distribution in which the 'send' operation is directed at a remote user profile (DMS).

Likewise, all production systems using FTP must have the development system libraries correctly configured for the OMS application. This allows production systems to send back responses directly to a known development system library. As a consequence of using the configured system data library for FTP distribution, an additional prerequisite of the FTP setup is that the message queue QRCVR must be found in the configured system data library, usually OMSDTA. If this is not the case, ensure that the object QRCVR/*MSGQ in OMSOBJ is moved into the system data library OMSDTA. This must be done at both development and production systems that are to use FTP distribution.

5.16.1.2 Changes to system configuration

New fields have added to the system configuration screen to accommodate the extra information required by FTP. The new fields are shown in bold on the following panel.

THN170 SEE/Change Development E Work with Systems De	nvironment tails
System Code SY1 System description SY1 Central DEV Machine Serial Number 01 System Type *AS400 OS/400 Version V 4 R 4 M 0	*SYS38/*AS400 Version/Release/Mod lvl
Communications Method <u>*FTP</u> SNADS User Id (P). <u>FTP</u> <u>SY</u> FTP Host name or IP address. <u>THN170</u> FTP User Id / Password <u>QSECOFR</u> / FTP SEE/Change Dev site code . <u>SI1</u>	*SNADS/*FTP 1
F1=Help F3=Exit F4=Prompt F9=Cmd F12=Cance	l F14=Ping FTP

5.16.1.3 Description of new fields

The following paragraphs describe the new fields.

Communications method

Enter the type of communication that will be used to send data to the system concerned.

FTP Host name or IP address

Enter the name of the system to which you wish to communicate via FTP. To use the host name the name must be set up in the AS/400's host table (option 10 in CFGTCP). If the target system is not configured as a host then simply enter the IP address in the usual format (eg. 123.456.789.123). You can use *F14=Ping FTP* to validate the address. However be aware that a succesful ping does not validate the password (See below) or even that the system pinged is an AS/400.

```
FTP User ID / Password
```

SEE/Change needs to log on to FTP before it can perform any functions. Supply an appropriate profile and password here, you must press Field Exit to clear the cursor from the field, otherwise the password encryption routines will not function uniformly. The password is not visible or depicted in any way.

FTP SEE/Change Dev site code

In association with the *prerequisites* section above, enter the site code where the OMS database can be found. This is resolved on the local configuration to obtain the target OMS database library.

When FTP is configured the SNADS user ID fields will default to 'FTP' and the system code. This is used internally and cannot be changed.

5.16.2 Configuration Summary

Once configured within the system configuration there are no further changes to make and no other SEE/Change panels have been altered. However there are some subtle operational differences between SNADS and FTP that are described below.

5.16.2.1 Operational Differences

FTP communications do not make use of distribution queues like SNA. Therefore unlike SNADS a file sent via FTP is despatched immediately rather than placed on a queue. SNA features recovery facilities so if a file is sent you can be fairly sure it will arrive, you just won't know exactly when. SNA may make several attempts to recover failed transmissions. FTP on the other hand is immediate, when the send command is issued, connection with the remote machine is established and the file sent straight away. If the send fails for whatever reason then the process is terminated.

Likewise in SEE/Change if a connection cannot be established and the file sent immediatley, a send error will occur (*SNDERR). Operator action will be required to take the SEE/Change option to recover from error where upon SEE/Change will try again.

5.17 Overrides (4.4200)

5.17.1 Object Type Override for a CR Type

SEE/Change allows the configured library to which an object is delivered to be overridden to a user-definable override library. The user can define three types of override:

- By Object a specific object will be delivered to a given library;
- By Object Type all objects of a given type (e.g *PGM) will be delivered to a given library

 By CR Type – all objects in a CR of a given type (e.g *EMG) will be delivered to a library.

CR Type overrides currently cause all objects within a CR to be promoted to the same library(s), regardless of what type of objects they are. To overcome this restriction, the current *Work with Application Overrides* function has been enhanced to allow the user to define Object Type Overrides that will only apply for a nominated CR Type.

Where more than one level of override could apply to an object, the priority of the overrides would be:

- By Object
- By Object Type for CR Type
- By CR Type
- By Object Type

The existing *Work with Application Overrides – by CR Type* screen has been amended to include a new function key (F14=Object types). This is a cursor sensitive function key. To access the object type overrides place the cursor on the line for the required CR type and press F14. The *Work with Application Overrides – by Object Type within CR Type* screen will then be displayed for the chosen CR Type as below:

THN170 SEE/Change Version 4.4200			
Work with Application Ove	rrides - by CR Type		
Application · AP1 Demo applicati	on 1 System: SV1 Centra		
Object Crown	on i bybeem, bii eeneid		
Object Group : Obj: Base applicati	.011		
Number of target libraries per CR type: 1			
	Overriding target libraries		
CR Type	Live/Prod Accept/OA Integr/Tst		
Program Bug Fixing	,, ,, , ,		
CD Classel as development			
CR Closea - no development			
Emergency Fix			
Application Modification			
New development			
Procedure Error			
riocedure milor			
User Error			
Program modification due to x-references			
	Bottom		
P2 Puit P0 and P12 General P12 Preset P	114 Object turner TOA Masser		
F3=EXIC F9=Cmd F12=Cancel F13=Repeat F	14=Object types F24=Messages		

SEE/Change – Supplementary Manual 4.5003

THN170 SEE/Ch	nge Version 4.4200				
Work with Application Overrides - by Object Type within CR Type					
Application : API Demo	oplication I System: SYI Centra				
Object Group : Obj: Base application					
UR Type : Emergency	1X				
Number of target libraries per ob	rriding target libraries				
Obi Tupo Attributo Livo/R	ad Accont/OA Intogr/Tet				
*BNDDIR	Ja Recept/on Integr/150				
*BNDDIR					
*CMD					
*CMD					
*CMD NOSRC					
*CMD NOSRC					
*DTAARA SQLRPT					
*DTAARA SQLRPT					
*DTAARA SQLVIEW					
*DTAARA SQLVIEW					
*FILE ADKBAT					
*FILE ADKBAT					
	More				
F3=Exit F9=Cmd F12=Cancel F13=R	peat F24=Messages				

5.17.2 Object Overrides

The introduction of multiple object override libraries in PE 4.4003 resulted in some anomalies in the *Work with Object Override Details* screen dialogue. These have now been resolved in PE 4.4200.

SEE/Change Version 4.4200						
Work with Object Override Details						
Enter the movement override specifications for object: Application: AP1 Demo application 1 Object: JSPF1 Type/Attr: *FILE PF Level: *BAS Number of target libraries per object: 1 Locate System/Site(P):						
System/Site : SY1 SY1 Centra SI1 SI1-Dev fo (Dev Site) Current target library sequence : 01						
Recom- Inc						
Config. Library pile? Jobd Data?						
Y APIJOBD N						
Live/Production Lib : APISYLLIV *CONFIG						
Acceptance/QA LDD : APISIACE ^CONFIG						
Load Source to Release Packet ? : Y Source pool library : APISY1POOL Source pool file : QDDSSO1 More						
F1=Help F3=Fyit F4=Prompt F9=Cmd F12=Cancel F23=Dlt overrides F24=Msgs						

The options for overriding Recompile, Job Description and Include Data? have now been mapped to each library. Defaults are shown directly beneath the subfile headings.

The record for Target Library sequence 01 defaults the library names to a new keyword "*CONFIG" which indicates the configured libraries (the library names are shown alongside). This allows amendment to the default Recompile/Jobd/Include Data settings without having to re-enter library names, and also means that the configured library names are not hard-coded in the override, so that if these were to change the overrides need not be amended. *CONFIG is only supported at Target Library sequence 01, as higher sequence numbers are used to deliver to extra libraries, not to override the configured libraries.

The keyword "*BYPASS" is now supported in the multiple override environment. Again this keyword can only be entered (when permitted) at Target Library Sequence 01. All other Target Library Sequences must have a blank library name for the by-passed environment.

(i.e. entering *BYPASS as the Mdl/Integration library means that this object will not be delivered to any library when promoting to Mdl/Integration).

The source overrides remain per object for a system/site (these can only be entered for target library sequence 01). The defaults are shown alongside.

Note: that there is an anomaly with ILE objects because SEE/Change registers the source, module and program separately. Only source overrides can be applied to the source (RPGLE_SRC etc), while library overrides must be applied to the objects (RPGLE_MOD and RPGLE etc).

5.17.3 Support for production systems pre - 4.4200

With PE 4.4200, extensive changes were made to the way SEE/Change handled object overrides. A new keyword *CONFIG was introduced which allowed an override to refer to configured libraries without having to re-type library names. This keyword is supported at both the development centre and production sites that have upgraded to 4.4200. However, it has been found that for customers wishing to upgrade their development centre without upgrading production sites, distributed object overrides no longer work correctly at the production sites.

To support the different behaviour of overrides pre and post 4.4200, *Work with Systems Configuration*, (WRKSYSCFG), has been amended to include the SEE/Change version number. When packaging a Release at the development centre, if all the target systems have been configured as SEE/Change Version 4.4200 (or higher), the *CONFIG keyword will be used when distributed overrides are sent. If one or more target systems is configured as a pre 4.4200 version (or the version number has not been entered), then *CONFIG will be replaced with the library name for that system (as configured at the development centre). Note that this will apply to all target systems for that Release (even if some of them are configured at 4.4200 or higher). It is recommended that the system configuration is visited on your Development system after upgrading to 4.4201 and that the correct SEE/Change version information is entered against each configured system.

THENON SEE/Change Development Environment				
Work with Systems Details				
System Code	SY3 SY3 Test Product AA9999 *AS400 V 5 R 1 M 0 4.4201	ion System *SYS38/*AS400 Version/Release/Mod lvl 4.4001 - 9.9999		
Communications Method SNADS User Id (P FTP Host name or IP address. FTP User Id / Password FTP SEE/Change Dev site code	*FTP). FTP SY3 . THENON . DMS / . SI3	*SNADS/*FTP		
F3=Exit F4=Prompt F9=Cmd F1 F17=DMS submit RCVRLS override	2=Cancel F14=Pin s	g FTP		

5.18 Authorise Release Distribution (4.4200)

The ability to control which users can send releases to a given remote system has been introduced in PE 4.4200.

The feature is enabled by setting the parameter @AUD – Authorise Distribution to Systems? to '*YES' using Work with Parameter Data.

A new function key (F16 = Distribution Authorisation) has been added to the *Work with Systems Details* screen in *Work with Systems Configuration*. Pressing this will bring you to the *Authorise Users for Distribution* screen. To authorise users to send to Live/Acceptance for this system, you must select a 'Y' in the appropriate column. Leaving a field blank implies the user cannot send to this environment. The list of users that appear in this screen is derived from the *Work with User Enrolment* function.

SEE/Change Version 4.4200 Authorise Users for Distribution					
System : EUR Europe Head Office					
Type "Y" to auth system. Press E	orise Users to Inter.	distribute to 3	Live and/or Accepta	ance for this	
	Distribute	Distribute			
User	to Live	to Acceptance			
AJONES					
DPRICE					
DTAYLOR					
GHURST	-	-			
LSMITH					
MSTEWART					
QPGMR	_	Ŷ			
RHARRIS		_			
SWRIGHT	Ŷ	Y			
TTOMKINS					
TWARNER					
VARCHER					
	_	_		More	
F12=Previous					

5.19 Automatic Release Handling (4.4200)

A new configurable option (Send to Live) has been added to the *Work* with Change Requests function in PE 4.4200. When this option is entered against a CR, a Release will be automatically created, the CR will be allocated to that Release and Release distribution will be initiated.

You can configure this option to control whether the CR is also simultaneously promoted to Live on the development machine. Alternatively, you can configure the option so that SEE/Change waits for a command from the production machine to trigger promotion to Live on the development machine (i.e. after the CR has been successfully received and installed at production).

The feature is enabled by setting the parameter @*SNL* – *Send to Live?* to '*YES' using *Work with Parameter Data*. Once this parameter is set, a new option - 48=Send to Live - will appear on the *Work with Change Requests* screen. This option may only be used against CRs with a status of Rdy/Rlease.

As part of the processing behind this option, a new command SNDCR is issued. This command includes two parameters:

- ENV determines whether the release is sent to ACP or LIV at production.
- PROMOTE determines whether the CR is also promoted to Live at development.

Use CHGCMDDFT to set this command to run correctly for your environment (the defaults are ENV(*LIV) and PROMOTE(*YES)).

If you want to configure this option so that an install of a CR to Live at production issues a command to trigger a promotion of that CR to live at development, do the following:
At the development system, change the command defaults for SNDCR to PROMOTE(*NO).

Create/modify an application *AFTER program and deliver this to the production system. Ensure this program contains the following code (this is included in the supplied template).

```
/*
    The following section of code will cause a transfer job (O#MVT)
                                                                         */
/*
                                                                         */
     to be sent back to the development system on the successful
/*
                                                                         */
    installation of a CR to Live. When the transfer job reaches
/*
    the development system, it will trigger a command to promote
                                                                         */
/*
                                                                         */
     the CR to live at development (MOVCRLIV).
/*
            ΤF
                       COND((&SYSM *NE &IRSY *AND &MOVT *EQ '*LIV') + */
/*
                        *OR (&SYSM *NE &IRSY *AND &MOVT *EQ '*ILV')) + */
                                                                         */
/*
                         THEN (DO)
/*
            RTVDTAARA DTAARA (CONFIG (11 10)) RTNVAR (&CFDTAL)
                                                                         */
/*
                                                                         */
            OVRDBF FILE (XRQ) TOFILE (&CFDTAL/XRQ)
/*
                                                                         */
            UPDO#MVL SYSM(&IRSY) IRNBR(&IRNO) CRSEQ(&SEQN)
/*
                                                                         */
            STRTFRRQS RQSN(O#MVL)
/*
            DLTOVR FILE (XRQ)
                                                                         */
/*
                                                                         */
            ENDDO
```

5.20 Overrides for DMS submit RCVRLS (4.4201)

5.20.1 Introduction

SEE/Change PE 4.4201 provides a new facility to allow greater control of the way a Release is received at a remote site. By default, the DMS job at a remote site will submit the RCVRLS using the configured application job description for the OMS application (usually OMSJOBD in library OMSDTA). The new feature allows you to override this job description at the remote site with your own job descriptions and / or scheduled dates / times. When you configure the overrides, each override is given an ID number. By assigning a corresponding number to the Release when sending it from the Development Centre, the DMS job will use the override job description (and / or schedule date / time) rather than the configured default when installing the Release.

A new general parameter @ROV (*Allow overrides for DMS submit RCVRLS?*) has been introduced which controls whether this new feature is available or not. Using *Work with Parameter Data*, set this parameter to *YES if you wish to use the new overrides (at both the development centre and the remote sites).

5.20.2 Configuring the Overrides for DMS submit RCVRLS

When the parameter Allow overrides for DMS submit RCVRLS? is set to *YES at a remote site, a new function key is available on the Work with Systems Details screen for the local system - F17=DMS submit RCVRLS overrides. Pressing this function key will display a pop-up window where up to nine overrides can be entered.

THENON SY	3 Test	Production S	vstem		
W	ork wi	th Systems De	tails		
System Code	: Co	nfigure Overr	ides for DM	AS Submit R	CVRLS :
System description	:	-			:
Machine Serial Number	: Type	override det	ails, press	s Enter.	:
System Type	:				:
OS/400 Version	: Ovr	Job	Schedule	Schedule	:
SEE/Change Version	: Id	Description	Date	Time	:
	: 1	OVRJOBD1			:
Communications Method	: 2	OVRJOBD2	<u> </u>		:
SNADS User Id	: 3		*SUN	01:00:00	:
FTP Host name or IP addr	: 4		*MON	01:00:00	:
FTP User Id / Password .	: 5		<u>*TUE</u>	01:00:00	:
FTP SEE/Change Dev site	: 6				:
	: /				:
	. 0				-
	• •				Bottom :
	• F12=	Cancel			
	:	Guilder			
F3=Exit F4=Prompt F9=Cmd	F12=C	ancel F14=Pi	ng FTP		
F17=DMS submit RCVRLS overr	ides		-		

The Override Id number is the number that will be matched by an incoming Release to determine whether an override is to be used.

If a job description is entered as an override, that job description must be created in the OMS data library (usually OMSDTA). If it does not exist and an incoming Release is expecting to find it, the DMS job will submit the RCVRLS using the default job description.

The schedule date can either be a valid date using the current date format (e.g. DD/MM/YY) or one of the special values: *CURRENT, *MONTHSTR, *MONTHEND, *MON, *TUE, *WED, *THU, *FRI, *SAT, *SUN.

The schedule time can either be a valid time using the current time format (e.g. 12:00:00) or the special value: *CURRENT.

5.21 System Configuration (4.4202)

5.21.1 Long Password Support

5.21.1.1 Introduction

With PE 4.4202, SEE/Change now supports OS/400 passwords of up to 128 characters. These new longer passwords were introduced in OS/400 V5R1M0 – the old limit was 10 characters. The availability of the new longer passwords is controlled by a new system value QPWDLVL. If this is set to 2 or 3, the longer passwords are used; if set to 0 or 1, the old 10 character passwords are used. The old 10-character passwords are uppercase and must be alphanumeric. The new 128-character passwords can be mixed case and can contain any character including blanks.

Note: SEE/Change will not support new longer passwords that contain the double-quote character (").

5.21.1.2 Work with Systems Details

The *Work with Systems Details* screen has been amended to allow for the longer passwords. If the system setting QPWDLVL on the current machine is set to 2 or 3, the screen will allow input of a longer password. Otherwise the password will be limited to 10 characters.

THNSPT SEE/Change Development Env	rironment
Work with Systems Deta	nils
System Code	*SYS38/*AS400 Version/Release/Mod lvl 4.4001 - 9.9999
Communications Method	*SNADS/*FTP
F3=Exit F4=Prompt F9=Cmd F12=Cancel F14=Ping F17=DMS submit RCVRLS overrides	J FTP

5.22 Authorisation (4.4202)

5.22.1 Authorisation for Users to Redevelop a CR

Thenon Authorisation Lists (TALs) provide a mechanism to ensure that nominated users explicitly authorise requests to promote a CR. Previously, the movement types that could be subject to TALs were:

*MDL	Promotion to Module/Integration at the Development Centre
*ACP	Promotion to Acceptance/QA at the Development Centre
*RDY	Change the CR status to Ready for Release
*LIV	Promotion to Live/Production
*ALC	Allocate CRs to a release
*RLS	Release to a Production environment

With PE 4.4202, a request to redevelop a CR (movement type *RDV) has been added to this list, so that it can also be subject to TALs.

5.23 SQL Support (4.4300)

5.23.1 Introduction

With the introduction of PE 4.4300, SEE/Change now directly supports SQL. Previous support for SQL in SEE/Change was via Database Management Commands where SQL statements can be attached to a traditional AS/400 file objects that had been retrieved into a CR and run when the CR is promoted/reverted.

The SQL language can be divided into two parts:

- Data Definition Language (DDL)
- Data Manipulation Language (DML)

DDL is concerned with the management of SQL-based objects and contains commands such as CREATE TABLE, CREATE INDEX, ALTER TABLE etc. most of which have equivalents in OS/400 (CRTPF, CRTLF etc).

DML is concerned with the data in files and contains commands such as SELECT, UPDATE, INSERT etc. (these commands have similar counterparts in traditional HLLs – e.g. READ, UPDAT, WRITE etc. in RPG).

With the introduction of PE 4.4300, SEE/Change now directly supports objects created using SQL DDL source. These SQL-based objects will participate in the change management cycle in much the same way as any other source-based object – the source can be retrieved to a CR, the object created, promoted/reverted, packaged and sent to a production machine.

For SQL tables, SEE/Change also supports the ALTER TABLE statement. This allows you to change the column layout and/or constraints on a table in one simple statement without having to worry about dependent logicals, constraints etc.

Also with PE 4.4300, SEE/Change will allow a script of SQL DML statements to be included in a CR. This script will then be run at each environment when the CR is promoted/reverted.

The following new SQL-based items are supported:

SEE/Change Type	SEE/Change Attribute	SQL Object	Description
*FILE	SQLTABLE	Table	A non-keyed single-member physical file
*FILE	SQLINDEX	Index	A keyed logical file scoped over a table
*FILE	SQLVIEW	View	A non-keyed logical scoped over one or more tables
*PGM	SQLPROC	Procedure	A program registered to a database
*PGM	SQLFUNC	Function	A service program that can be referenced in SQL
*PGM	SQLTRIGGER	Trigger	A program attached to a table that may be fired when the table is accessed.
*PGM	SQLDML	-	A script of SQL DML statements

The new SQL object types have been defined in XOT along with two new source usage values:

- *RUNSQLSTM (Used for SQL DDL)
- *RUNSQLDML (Used from SQL DML)

5.23.2 Configuration

5.23.2.1 Basic configuration

New object types (and default source files) have been defined in SEE/Change for the new SQL items. In order to use SQL, it will be necessary to create appropriate source files and register them to SEE/Change using *Work with Application Source Pools* in *Work with Application Configuration*. The source files must have a record length of 92 characters.

THNSPT		SEE/Change Work with A	Developmen Application	t Environme Source Poe	ent ols		
Application	n	: AP1 Demo application 1 Local Sys. :				DEV	
Source pool	1 id	: <u>*BAS 01</u> Source for Primary Base pool Mor					
Object Type *FILE *FILE *FILE *FILE *FILE *FILE *MSGF *MSGF *MSGF *PGM	Attribute PF38 PRTF PRTF38 SQLINDEX SQLTABLE SQLVIEW UIM MNUCMD MNU36 MSGF36 ADKBAT	Usage *COMPILE *COMPILE *RUNSQLSTM *RUNSQLSTM *RUNSQLSTM *COMPILE *COMPILE *COMPILE *COMPILE *COMPILE *INTERPRE2	Source File QDDSSRC QDDSSRC QDDSSRC SQLINDSRC SQLVIEWSRC QMNUSRC QMNUSRC QS36SRC QS36SRC QADKBAT	Source Library AP1DEVSRC AP1DEVSRC AP1DEVSRC AP1DEVSRC	Patrn Table	Моз	re
F1=Help F1	3=Exit F4=	Prompt F7=1	Prev pool	F8=Next poo	ol F9=C1	md F12=Cancel	IS
F13=Rpt F1	16=Update	F17=Clear Po	bol F23=Ta	rget Pools	for new	mbrs F24=Msc	

5.23.2.2 Registering existing source

If you have existing SQL source that you wish to register to SEE/Change, you can copy members into their respective source files as configured your application source pool library and execute the UPDSRCREG command to complete the registration process. The following table describes the *default* source files and source types that SEE/Change is configured to use:

SEE/Change Attribute	SQL Object	Default SEE/Change Source File	Default SEE/Change Source Member Type
SQLTABLE	Table	SQLTBLSRC	ТХТ
SQLINDEX	Index	SQLINDSRC	ТХТ
SQLVIEW	View	SQLVIEWSRC	ТХТ
SQLPROC	Procedure	SQLPROCSRC	ТХТ
SQLFUNC	Function	SQLFUNCSRC	ТХТ
SQLTRIGGER	Trigger	SQLTRGSRC	ТХТ
SQLDML	-	SQLDMLSRC	SQLDML

In all cases:

- The source must not qualify object names with schema (library) names.
- DLL source members must only contain source to create one object and that object must have the same name as the source member and the object created must be of the correct object type.

For example MYTABLE in SQLTBLSRC might contain the SQL:

create table MYTABLE (MYCOLUMN char(10) not null
with default);

5.23.2.3 SQL Schemas

A schema is a collection of SQL objects. OS/400 implements a schema as a library that contains a journal together with local views of the system catalogue. Any files added to the schema are automatically journalled and appear in the local catalogue views.

On the *Work with Application Details* screen in *Work with Application Configuration*, a new flag has been introduced to influence the behaviour of CRTCRLIB in respect of this new subtlety:



The new flag defaults N(o) (i.e. SEE/Change will create CR work libraries as ordinary libraries *not* schemas). Set this flag to Y(es) if you know that you want most of the CR work libraries for this application to be schemas; for instance, if your application uses commitment control and you test changes to files while working in your CRs.

The setting for this flag applies application wide, but you can override the value when creating individual CRs.

	THENON Change Mar Work with Change	nagement system. Request Details
Change Application	.(P). V43	V4.4300 application
Request Summary Text . CR Type Contact Reference	SQL develop .(P). *NEW 	oment CR New development
CR CASE Tool Estimated Hours	.(P). *VH1 .(P). *NONE 	very High
Assigned User/Grp Profi Planned concurrent dev Library List level	le . QPGMR ? *NO .(P). *BAS	*YES/*NO Application Base Level
Retrieval Restriction CRTCRLIB to create sche	*NO ema *NO	*YES/*NO/*LVL *YES/*NO
F1=Help F3=Exit F4=Pr	compt F9=Cmd F12	2=Cancel

The default value on this panel is derived from the application definition.

5.24 Bulk Management of Database Overrides (WRKAPPOVR) (4.4301)

It is now possible to manage all overrides for database objects from a single new display.

The new panel is accessible through the command WRKAPPOVR or from the *Work with Application Configuration* display option *9=Overrides*:

THNSPT Change Management System Work with Application Configuration						
2=Change 9=Overrides	3=Copy 12=Where used	4=Delete 14=Src files	5=Display 16=CASE info	6=ASSET Cfg 17=JDE info		
Opt App Descr < L 9 V43 V4.43	iption ocate 01 Demo applicat:	Lon				
F1=Help F3=Ex	it F4=Prompt F5 enon app F23=Mon	5=Refresh F6=Cr	eate F9=Cmd F1 Messages	Bottom 2=Cancel		

Once selected the Work with Bulk Overrides display is invoked:

		Change Mana Work with B	gement system ulk Overrides		
System: P43	Production V	1.4300	Application: V43	V4.4300 test ap	plication
Type options 2=Change	, press Enter 3=Copy 4=Dele	ete 7=Copy -	based on Library	7	
Opt Object	Туре	Attrib			
ITMMST	*FILE	PF			
F3=Exit F5=	Refresh F6=C	reate F9=Com	mand F12=Cancel	F13=Repeat	Bottom

Here, all overrides for database objects for the local system and selected application are shown. Option *2=Change* allows modification of the overrides so that refinements can be made. The resulting display is arranged so that all the possible override target libraries for each of the configured sites / environments may be modified:

Change Management systemMode: ChangeWork with Bulk Overrides						
System:P43 ProductionV4.4300Application:V43 V4.4300 test applicationObject:ITMMSTType/Attribute:*FILE / PF						
Type options, press Enter. 2=Change Extended Overrides	Trut					
OptLive/ProdAccept/QAIntegr/TstSite/DescriptionCTLLIVCTLTSTGATGatwick*BYPASS*BYPASSGLAGlasgow*BYPASS*BYPASSLIVLiverpool*BYPASS*BYPASSLONLondon	Ovr					
F3=Exit F5=Refresh F9=Command F12=Cancel	Bottom					

In the above example the Item Master file (ITMMST) is being overridden so that it is delivered to a single library named CTLLIV and CTLTST, for the live and acceptance test systems respectively, instead of the configured library for the Gatwick site. The remaining sites make use of the *BYPASS special value, indicating that the file is not to be delivered to these sites at all. This display also supports the *CONFIG special value. Because this configuration is a production system there is no Integration test environment configured.

Further extended override options may be made using option 2=Change *Extended Overrides*: If extended override settings have previously been set, this is indicated on the above panel by a "Y" appearing in the "Ext Ovr" column.

	Change Management system Work with Bulk Overrides	Mode: Change
System: P43 Production V Object: ITMMST Typ	4.4300 Application: V43 V4.4 me/Attribute: *FILE / PF	300 test application
Type options, press Enter 2=Change Extended Overr	ides	Rest
Opt Live/Prod Accept/Q 2 CTLLIV CTLTST *PVDASS *PVD	A Integr/Tst Site/Description GAT Gatwick	Ext Ovr
<u>*BYPASS</u> *BYP:	Extended Override Details	:
	<- Recompi Y/N Job D	le -> Inc : pesc Dta :
:	Live/Production : Acceptance/QA : Mdl/Integration :	
: : F	3=Exit F9=Command F12=Previous	
: F3=Exit F5=Refresh :		: :

Pressing <Enter> as necessary will confirm changes and return control to the main *Work with Bulk Overrides* panel from where other options may be executed.

Option *3=Copy* allows the overrides for one object to be replicated to another that does not already have overrides. If overrides do already occur for the target object name / type / attribute an error occurs.

Option 7=*Copy based on library* allows the override for one object to be replicated to a list of other object names where the list is prepared from a nominated user library. In this case object names that already have overrides are overwritten with the new settings.

For example, in our factitious example the ITMMST file is one of a small number of common control files shared by all of the sites. To have the same override settings applied to all the other files option 7 is used:

Change Management system Work with Bulk Overrides							
System: P43 Production V4.4300 Application: V43 V4.4300 test application	Syste						
Type options, press Enter. 2=Change 3=Copy 4=Delete 7=Copy - based on Library	Type 2=(
Opt Object Type Attrib	Opt						
7 ITMMST *FILE PF	7_						
: Copy - based on Library :							
: Enter Library Name: CTLLIV :							
: :							
: F3=Exit F9=Command F12=Previous :							
F3=Exit F5=Refresh F6=Create F :	F3=E>						

Given a library name SEE/Change will build a list of all the objects in that library that are eligible for database overrides and present a confirmation display where the user may refine the selection if necessary:

	Change Management system Mode: Work with Bulk Overrides Mode:								
Syst Obje	System:P43 ProductionV4.4300Application:V43 V4.4300 test applicationObject:ITMMSTType/Attribute:*FILE / PF								
Type X=	Type options, press Enter. X=Select for copy								
Opt	Name	Tvpe	Attribute	Text					
X	CSTMST	*FILE	PF	Customer Master File	(PF)				
x	CSTMSTL1	*FILE	LF	Customer Master File	(LF)				
x	CSTMSTL2	*FILE	LF	Customer Master File	(LF)				
x	CSTMSTL3	*FILE	LF	Customer Master File	(LF)				
x	ITMMST	*FILE	PF	Item Master File (PF)					
X	SHPMST	*FILE	PF	Shipment Master File	(PF)				
x	SHPMSTL1	*FILE	LF	Shipment Master File	(LF)				
					Bottom.				
E2-E	wit E5-Dof	roch EQ-Com	mand E12-Car						
r S-E	AIL PD-REL	Lesii F9=Colli	nanu r12–Cai	ICET					

Pressing <Enter> will propagate the override settings for ITMMST to each of the other objects:

SEE/Change – Supplementary Manual 4.5003

	Change Management system Work with Bulk Overrides									
Syst	em: P43 Pro	duction V	4.4300	Appli	cation: V4	3 V4.4300 tes	t application			
Туре 2=	Type options, press Enter. 2=Change 3=Copy 4=Delete 7=Copy - based on Library									
Opt	Object	Туре	Attrib							
	CSTMST CSTMSTL1 CSTMSTL2 CSTMSTL3 ITMMST SHPMST SHPMSTL1	*FILE *FILE *FILE *FILE *FILE *FILE	PF LF LF PF PF LF							
F3=E	Xit F5=Ref	resh F6=C1	reate F9=0	Command	F12=Cance	l F13=Repeat	Bottom			

5.25 Suffix for Restore of DMS library (4.4301)

5.25.1 Introduction

When a release library is sent via COMMS, it will be physically packaged into a save file. When this save file is received and unpacked at a target site, it is restored with a name in the format AAA999999, where AAA is the development system name and 99999 is the transfer number.

If distributing to more than one logical production system (OMSDTA data set) on the same machine, then a conflict will arise in the communications manager, and all but one of the software releases will fail.

With PE 4.4301, a new parameter – Suffix for restore of DMS library (@SDL) – has been introduced to allow a unique DMS library name to be defined at each target system, so that the conflict is avoided and the restores can run concurrently.

5.25.2 Configuration

The new parameter (@SDL) is accessed via the *Work with Parameter Data* (WRKPRMDTA) screen.

SEE/Change – Supplementary Manual 4.5003

Change Management System Work with Parameter Data									
2=Change 5=Display									
Opt Code	Description	 VAR	Para LEN	DEC	er af DSP	ttril EDT	vLD	s AUT	
@OMR @OPM @OVM @PCK @PKG @PKG @RCH @ROV @SBM 2 @SDL	THENON Transfer Request Name Operator Message Queue Name Authorised usrprf for CR control objects DMS Initial Packet Size Limitation Release packaging default Incoming data transfer hold status Allow overrides for DMS submit RCVRLS? Job Description for submitted CR jobs Suffix for restore of DMS library		5 10 10 5 7 4 4 10 2	A A A A A A A A			V Y V V R	2 2 2 2 2 2 2 2 2 2 2 2	
	Shipment Libraries Outgoing data transfer hold status Send to Live from WRKCHGRQS Send Type default TAPE/COMS Site Specific Job Descriptions 4 Compile SEE/Change Panel Style	10	40 4 4 4 4 1	A A A A A			Y Y R Moi	2 2 2 2 2 2 2	
F1=Help	F3=Exit F6=Create F9=Cmd F11=Delete F	F12=0	Cance	el I	721=1	Prin	t li:	st	

Option 2 allows the user to maintain its' value.

	Change Management System Work with Parameter Data	:
Attr: *LEN: 0 / 2 A *DSP: *STD *EDT: *STD	*VLD: Value Range: *VL1: 01 *VL2: 99	• • •
Enter/update parameter v	alue: @SDL : Suffix for restore of DMS library	· ·
	Parameter Value SUFFIX (0199) 01	· · · · · · · · · · · · · · · · · · ·
		:
		:
		•
		Bottom .
F1=Help F3=Exit F9=Cmd	F12=Cancel	•

If this parameter is left blank, then a suffix is not used for the restored DMS library name. The restored DMS library will be in the format AAA99999, where:

AAA	=	Development System name
99999	=	Transfer Number

And is the SEE/Change default.

If a valid number between 01 and 99 is entered for this parameter, then the suffix will be used when determining the restored DMS library name. The restored DMS library will be in the format AAA99999XX, where:

AAA	=	Development System name
99999	=	Transfer Number
XX	=	Suffix for restore of DMS library

In order to allow the restores of the DMS library to run concurrently for different target systems on the same machine, a unique setting for this parameter should be entered in each target system.

5.26 Archiving and Work Library Management (4.4301)

5.26.1 Introduction

SEE/Change creates one library for each Change Request. It may also create none, one or many archive libraries as a result of a CR moving to live. Prior to PE 4.4301, two methods had evolved for managing these work libraries.

- Work with Library Status / SEEAUTO house cleaning job;
- Consolidated archive library.

For full descriptions of these functions please refer to the PE notes for the releases in which they were introduced (4.3006 and 4.4003 respectively). The following sections provide a summary of these library management systems.

5.26.2 Work with Library Status

This library management system was first introduced in 4.3006.

It introduced the WRKLIBSTS and the SEEAUTO commands.

WRKLIBSTS allowed the user to find out why a library existed on the system (which CR it related to) and when the CR had gone live. For CR's that had not yet gone live no such date can be given.

Within this screen it is possible to select expiry time periods for work libraries related to CR's that have already gone line. For instance is is possible to say "Keep CR work libraries for 90 days after the date that they go live, save them to tape then delete them". It is the SEEAUTO batch job that actually performs the housekeeping. Thus on running SEEAUTO after the 90 days has elapsed, work libraries meeting this criteria will be saved to take then removed.

5.26.3 Consolidated archive library

A second method was later introduced. This method aimed to reduce the number of CR work and archive libraries that were required on the system and employed a pack / unpack routine so that after going live, all the work libraries related to the CR were packed into save files of the same name in a single nominated archive library. Should reversion be necessary they necessary libraries were unpacked first.

5.26.4 **Pros and Cons**

Both systems allow for the purging of excessive archive levels, albeit through different methods.

The WRKLIBSTS / SEEAUTO approach is appealing because it allows all change related work libraries to be removed after a predefined time period has expired. *But*, it requires the libraries to remain on the system in an unconsolidated form.

The consolidate archive library approach is appealing because it allows all change related work libraries to be held in a packed form within *one* purposefully dedicated library. *But*, this consolidated library can become large and there is no way of purging save files from it using a date live criteria.

5.26.5 Amalgamated system

PE 4.4301 addresses these issues by amalgamating the functionality of both previous systems to provide one cohesive library management solution.

It is therefore now possible to configure consolidated archive libraries *and* use the WRKLIBSTS and SEEAUTO commands to purge save files from these based on a predefined ageing criteria.

5.26.6 Work with Library Status (WRKLIBSTS) display

Accordingly the WRKLIBSTS display has been changed to accommodate the new level of functionality:

Change Management System Work with libraries							
Timo onti	na nroga I	Intor			Res	ident Libraries	
1=Save 2=Change 4=Remove entry 5=Display							
Date				Override			
t Live	Library	Type	Related CR	Action	Status	*LIB/*SAVF	
0/00/00							
0/00/00	W#00000205	CR	000002 / 05	5	Active	*LIB	
0/00/00	W#00000206	CR	000002 / 00	5	Active	*LIB	
0/00/00	W#00000207	CR	000002 / 0	7	Active	*LIB	
8/03/04	W#00000201	CR	000002 / 02	L *NONE	Active	*SAVF	
8/03/04	W#00000202	CR	000002 / 02	2 *SAVDEL	Active	*SAVF	
8/03/04	W#00000203	CR	000002 / 03	3	Active	*SAVF	
8/03/04	W#00000204	CR	000002 / 04	1	Active	*SAVF	
8/03/04	W00000010	Archive	000002 / 02	2	Active	*SAVF	
8/03/04	W00000011	Archive	000002 / 02	2	Active	*SAVF	
8/03/04	W00000012	Archive	000002 / 03	3	Active	*SAVF	
						More	
B=Exit FS	5=Refresh 1	9=Commar	nd F10=Res:	ident/Non	Resident	F12=Cancel	
6=Contro	l Options I	18=Rebui	ild Informat	cion			
	-						

WRKLIBSTS now show:

- The CR to which the work library is related;
- The status of the work library, *SAVF or *LIB, depending on whether it is packed into an archive library or not;
- The override action, if specified;

The other panels in WRKLIBSTS have not changed.

5.26.7 SEE/Change Automatic clean up (SEEAUTO) command

The SEEAUTO job has been enhanced to first determine if a library if packed or not, and then execute appropriate commands depending on whether it has been packed or not.

5.26.8 Recommended implementation

The following bullets are recommendations to the implementation of library management. Please use these in conjunction with previous release notes to understand planning and preparations steps such as packing / unpacking and archive level purging. The recommendations are based on the following assumptions:

- Most sites prefer to use the consolidated archive library method;
- Most sites have a backup cycle that encompasses all the SEE/Change libraries and so only require SEE/Change to perform the purging aspect of the library management functionality;
- Most sites wish to be able to revert for a fix period of days (for example 45) after which they commit to forward fix, and therefore do not require work libraries.

If your site meets the above criteria, the recommended implementation is:

- Configure application archive libraries. The same library can be used for the entire system if preferred, else archive libraries can be configured by application;
- Pack existing work libraries into the archive library if not done already (See PE 4.4003);
- Access WRKLIBSTS, option F16=Control Options and specify the number of days you wish to keep work and archive libraries. Specify the same number of both and specify *DELETE for the action to take after expiry;
- Run SEEAUTO on a regular basis, say weekly, using the job scheduler.

5.27 Support for non-SQL trigger programs (4.4301)

This section applies only to existing triggers created using ADDPFTRG. SEE/Change provides native support for SQL triggers – see PE Notes 4.4300 for details on these.

In a typical application configuration, each environment will contain a database library and a program object library. Each environment's database library will contain a full set of files, but only the live environment's program library will contain a full set of program objects. Intermediate environment program libraries will only contain program objects that have been promoted to that environment by virtue of CR's that are not yet live.

This implies that a trigger defined on a physical file in any environment will reference the program in the Live program object library, as the program will not normally exist in any other environment. This is depicted in the following diagram:



However, if the trigger program is retrieved to a CR and then promoted to an intermediate environment, we need to realign the trigger at that environment (and any lower environments) to reference the new version of the program. For example, if out program "T" was changed and moved to the Acceptance testing environment, we would require trigger processing to be arranged as follows:



To achieve this, a new command *SETTRGPGM* (Set Trigger Program) has been introduced at PE 4.4301. This command will be called after each movement of a program. This command will firstly check the system catalogue to see whether there are any entries for the program as a trigger program that references a physical file in a configured database library for the current application. If so, the command will attempt to re-align the trigger at each configured environment. The initial library list of the application job description of each configured database library is searched to find the current location of the trigger program. If this location differs from the catalogue location, the existing trigger is replaced so that it references the new program library. The trigger will then point to the promoted version of the program.

An error in this processing will fail the movement and diagnostic messages will be listed to the movement log. The movement will get an error status.

5.27.1 Configuration

The new support for non-SQL triggers can be enabled on a perapplication basis using *Work with Application Configuration*.

THNSPT Change Managem	ment System
Work with Applica	ation Details
Document Processing Enabled N	(Y)es/(N)o
Document Top Level Folder AP1	
ILE Processing Enabled Y	(Y)es/(N)o
ILE Remote Delivery Method PGM	PGM/MOD
ILE Module Pool Library AP1DME	DLPOL
ILE Allow missing modules Y	(Y)es/(N)o
Cross Referencing Utilisation *FULL	*NONE/*DATA/*FULL
JDE Application? N	(Y)es/(N)o
SEE/Change Version for JDE 00	(00 - 99)
Development Prod. System(P).	
Development Prod. Site(P).	
Unlock Source at Dev-Prod System . N	
IFS Processing Enabled Y	IFS Archive Library AP1DARCLIB
IFS CR No. of Archive Levels 00	
CRTCRLIB to create schema? N	(Y)es/(N)o
Maintain existing triggers? Y	(Y)es/(N)o
	P
	Bottom
E1-Holp E3-Evit E4-Prompt E9-Cmd E12-	Cancol E16-Undato E24-Mossagos
FI-Heip FS-Exic F4-Fiompt F9-Cmd Fiz-	-cancer rio-opuace rz4-Messages
1	1

5.28 Take on Jobs Console (4.4304)

Often, SEE/Change is configured over an existing application. In this instance a number of take on jobs must be run before reliable Change Management may commence for the newly configured application. In this release a new panel has been introduced to help users know what job to run and in what sequence. The new panel is accessible via the Work with Application Configuration (WRKAPPCFG) function as follows:

THNSPT	THNSPT Change Management System Work with Application Configuration									
18=IFS Paths	30=Take On Jobs 40=Wrk auth 1st 43=IR dft schd 44=CR dft schd									
Opt App Descrip	ption cate									
<u>30</u> ACM ACME O	rder Processing									
	Bottom									
F1=Help F3=Exi F21=Include The	t F4=Prompt F5=Refresh F6=Create F9=Cmd F12=Cancel non app F23=More options F24=Messages									

Note that previously option 30 ran only the Update Source Register take on job (UPDSRCREG). UPDSRCREG is now incorporated into the new function.

The new panel is also accessible via the new Work with Application Take on Jobs (WRKAPPTOJ) command. The WRKAPPTOJ command accepts the application code as a parameter:

WRKAPPTOJ APPL(xxx)

When selected, the following panel is presented:

Change Management System Application Take on Job Console								
Application : ACM ACME Order Processing								
Type options, press Enter. 1=Run Job 5=Work with Result 8=Work with Last Job								
Opt Take on Job (run in sequence listed) Last Run Status								
1_ Build Source Register	(UPDSRCREG)	00/00/00	00:00	Not Run				
Build ILE reference information	(BLDILEREF)	00/00/00	00:00	Not Run				
Build Non-source based objects register	(UPDNSOREG)	00/00/00	00:00	Not Run				
Build Cross-Reference database	(REFXREF)	00/00/00	00:00	Not Run				
				Bottom				
F1=Help F3=Exit F5=Refresh F9=Command F1	2=Previous							

The panel shows all of the jobs that need to be run for a new application configuration in the sequence in which they should be run.

Note the introduction of the *Build Non-Source Based Objects Register* (UPDNSOREG) take on job. This is a new job that has been introduced in this PE and is documented later in this note.

Option 1 prompts the selected command for review and invites the user to submit the job. At this point the "Last Run" column is updated to show the date and time the job was last run.

Change Management System Application Take on Job Console									
Application : ACM ACME Order Processing									
Type options, press Enter. 1=Run Job 5=Work with Result 8=Work with Last Job									
Opt Take on Job (run in sequence listed)		Last Run		Status					
Build Source Register	(UPDSRCREG)	11/10/04	14:59	Queuing					
Build ILE reference information	(BLDILEREF)	00/00/00	00:00	Not Run					
Build Non-source based objects register	(UPDNSOREG)	00/00/00	00:00	Not Run					
Build Cross-Reference database	(REFXREF)	00/00/00	00:00	Not Run					
				Bottom					
Job 429304/SPENCER/UPDSRCREG submitted to job queue @JOBQCM in library QGPL. F1=Help F3=Exit F5=Refresh F9=Command F12=Previous									

Once a job is submitted the status will cycle through values of *Queuing*, *Active* and finally, when complete, one of the following completion statuses:

- Run/OK meaning the take on job has completed successfully;
- Run/Wrn meaning the take on job has completed successfully but with some warnings. In this instance the job output should be reviewed and corrective action taken if necessary;
- Run/Err meaning the take on job encountered errors during its processing. It
 has completed unsuccessfully. In this instance the job output should be
 reviewed, corrective action taken and the job rerun.

Option 8=Work with Last Job executes a WRKJOB command for the last run instance. The take on job will have generated a report that may be viewed from within this function.

Option 5=Work with Result is only available against the REFXREF job since it accesses a special display to aid in assessing and resolving problems with Cross Referencing. This panel is also newly introduced in this PE and is also documented later in this note.

5.29 Update Source Register (UPDSRCREG) (4.4304)

The Update Source Register (UPDSRCREG) process now generates a report, an example of which follows:

UPDSRCRI	EG THNSPT		Ch	ange Management S	ystem			10/28/04 17:18:29	Page :	1
			U	pdate Source Regi	ster					
				PARAMETEF	t S					
				Application Source File Source Library Source Type Object Type Object Attribute.	A	CM ACME	Order P	rocessing System		
	Source Library	Source File	Object Type	Object Attribute	Source Type	Source Qual.	Mbrs Procd.	Result		
Totals	ACMELIVSRC ACMELIVSRC	QCLSRC QCLSRC	* PGM	CLP	*BAS	01	10 10	*OK Note: 10 members in File		
Totals	ACMELIVSRC ACMELIVSRC	QCMDSRC QCMDSRC	*CMD		*BAS	01	2 2	*OK Note: 2 members in File		
	ACMELIVSRC ACMELIVSRC ACMELIVSRC ACMELIVSRC ACMELIVSRC	QDDSSRC QDDSSRC QDDSSRC QDDSSRC QDDSSRC	*FILE *FILE *FILE *FILE *FILE	DSPF LF MNUDDS PF PRTF	*BAS *BAS *BAS *BAS *BAS	01 01 01 01 01	11 9 2 6 3	*OK *OK *OK *OK		
Totals	ACMELIVSRC ACMELIVSRC	QDDSSRC QDDSSRC	* MSGE	MNUCMD	*BAS	01	33	*OK Note: 33 members in File		
Totals	ACMELIVSRC ACMELIVSRC ACMELIVSRC	QRPGLESRC QRPGLESRC QRPGLESRC	* PGM * PGM	SQLRPGLE_S	*BAS	01	17 0 17	^OK *OK Note: 17 members in File		
Totals	ACMELIVSRC ACMELIVSRC	QRPGREF QRPGREF	* PGM	RPGREF	*BAS	01	0 0	*OK Note: O members in File		
Totals	ACMELIVSRC ACMELIVSRC	QRPGSRC QRPGSRC	* PGM	RPG	*BAS	01	16 16	*OK Note: 16 members in File		
Source Me	ember Types Pro	cessed Rest	<u>ilt</u>	ata Sauraa Bagiat	or Comple	tod Norm				
	0	* OK * ERI * WAI	- Upd RORS - Err RNINGS - War	ors exist, review nings exist, revi	and reru .ew and re	n UPDSRC run UPDS	REG to eRCREG if	nsure registration complete necessary	s normally	
			<<<	End of Report >>	·>					

The report is organised by source file and, for each file, the member types that have been processed along with a count, are detailed. These are totalled.

Where the total number of members registered for a given file differs to the total number of members in the file, a warning is generated. In this instant it is likely that a member type is present in the file that has not been registered for use in SEE/Change.

Where a file is missing or locked or some other unexpected condition occurs, an error is generated.

Warnings and errors will require investigation and corrective action after which the take on job will require re-running.

5.30 Update Non Source Based Objects Register (UPDNSOREG) (4.4304)

Previously, SEE/Change only tracked non-source based objects once they had been promoted through the Change Management life cycle. The only exception to this is ILE programs; these being registered by the Build ILE References (BLDILEREF) process.

In PE 4.4304 a new take on job has been introduced that allows users to perform analysis over the configured application libraries and register all non-source based parts such as data areas, data queues, message files and so on.

This new function is accessible via the Work with Application Take on Jobs (WRKAPPTOJ) function as follows:

Change Management System Application Take on Job Console										
Application : ACM ACME Order Processing										
Type options, press Enter. 1=Run Job 5=Work with Result 8=Work with Last Job										
Opt Take on Job (run in sequence listed) Last Run Status										
Build Source Register	(UPDSRCREG)	11/10/04	14:59	Run/OK						
Build ILE reference information	(BLDILEREF)	00/00/00	00:00	Run/OK						
1_ Build Non-source based objects register	(UPDNSOREG)	00/00/00	00:00	Not Run						
Build Cross-Reference database	(REFXREF)	00/00/00	00:00	Not Run						
				Bottom						
F1=Help F3=Exit F5=Refresh F9=Command F1	2=Previous									

It is also accessible via the native command interface. The command is UPDNSOREG and accepts an application code and library name as parameters:

UPDNSOREG APP(xxx) LIB(xxxxxxxxx)

When selected, or prompted, the following panel is presented:

Up	odate Non-Source	ce Register ((UPDNSOREG)	
Type choices, press En	nter.			
Application	· · · · · · · · · · ·	ACM *ALL	Character value Name, *ALL	
F3=Exit F4=Prompt	F5=Refresh	F12=Cancel	F13=How to use this d	Bottom isplay
F24=More keys				-1 - 2

The application code is mandatory and must be valid.

The default value for the library parameter is *ALL. In this case all *live* configured libraries at base, site and group levels are inspected. Libraries at the acceptance and module levels are not inspected.

If a library name is specified, it must be a library that is configured at the live environment level for the specified application.

The processing for *each* library is, briefly, as follows:

- Identify the level to which the library belongs (Base, Site or Group);
- List library objects;
- Discard ILE derived objects (these will have been registered by BLDILEREF processing);
- Discard objects that have corresponding source (these will have been processed by UPDSRCREG);
- Register the remaining objects to the non-source based objects register file (O#NSO);
- If the object being registered is a message file, further special processing is necessary. This processing is described in more detail in the following sections.

An audit file is produced to show the objects that have been considered for registration along with a completion code. Errors should be reviewed and corrected and the procedure rerun. This process should be repeated iteratively until all errors are eliminated. Warnings may be eliminated or ignored once the implications have been understood.

There are some noteworthy points:

 Objects found in program libraries that have an incompatible type / attribute are disallowed. The exception is source files present for interpreted source file purposes (copy / include source, interpreted source etc);

- Objects found in data libraries that have an incompatible type / attribute as are disallowed;
- Typical example of invalid combinations (based on the configuration as shipped) are:
 - Physical and logical files in program libraries;
 - Program objects in data libraries.
- The rules governing what is allowed / disallowed are defined by the application configuration files maintained via WRKAPPCFG and the object types master file (XOT). Therefore, user defined object types are supported, provided their SEE/Change object type can be uniquely determined given their object type, attribute and library type.
- The non-source based objects register file is O#NSO.
- **WARNING:** In order for UPDNSOREG to achieve accurate results, The Update Source Register (UPDSRCREG) and Build ILE References (BLDILEREF) jobs <u>must</u> have previously completed successfully.

The following sections cover the handing of message files, starting with some background in this area.

5.30.1 Message File Handling – Pre 4.4304

In SEE/Change's Development Manager, message files are retrieved for maintenance into a municipal library. This library is known as the *Application Message File Library* and is configured, by application, in the primary Work with Application Configuration (WRKAPPCFG) panel. Accordingly, a message file may be retrieved into several CRs at once. However, concurrent maintenance shall result in the same message file being maintained; i.e. that in the application message file library.



A special SEE/Change editor is used to maintain message file message descriptions. This allows the developer to search message texts and copy message descriptions in addition to the native OS/400 message handling functionality. SEE/Change carries out the native message description handing in the background. In order for these functions to work, SEE/Change uses and maintains some internal files. These also reside in the application message file library and are called:

- XMG (Message file headers) and
- XMI (Message descriptions)

The first time a message file is retrieved into a CR a special registration process is run to populate XMG and XMI.

When a CR that includes a message file is promoted, it is the message file from the application message file library that is merged or duplicated, as per the application configuration, to the target and intermediate environments. Consequently the message file never exists in the CR work library at all. The CR entry is merely a pointer to the real message file in the application message file library and serves as a memo for the movement mechanism to perform message file processing when the CR is moved.

5.30.2 Message File Handling – Post 4.4304

Prior to 4.4304, the XMG / XMI registration process only occurred the first time a message file was retrieved into a CR. Furthermore, messages files never featured in the non-source based object register, even after having been promoted live.

In this PE the process of message file registration has been incorporated into the new UPDNSOREG processing.

When a message file is encountered in an application library, SEE/Change looks to see if the message file exists in the application message file library and the XMG / XMI registration files. It processes the message file according to the following table of conditions:

In Application's Object Library?	In Message File Library?	Action Taken
Yes	No	XMG and XMI are rebuilt <i>based on the message</i> <i>file found in the Application's Object Library</i> . The non-source based objects register is updated as necessary.
Yes	Yes	XMG and XMI are rebuilt <i>based on the message</i> <i>file found in the Application Message File library</i> . The non-source based objects register is updated as necessary.
No	-	None.

5.30.3 Re-registering a message file.

In the normal course of events the message file in the application message file library will be the most recent version. Accordingly it is this message file that is used in the registration process, if present.

If for some reason the version in the application message file library is not the most recent version, it will be necessary to refresh SEE/Change. To do this adopt the following procedure:

- Ensure the version in the configured applications' live library is the required version. Backup the version from the application message file library.
- From a command line, enter the following command:

DLTAPPMSGF APPL(xxx) MSGF(xxxxxxxxx)

This will remove the message file from the application message file library and the associated control files. It does not remove the message file history or registration information.

Next, create a CR for the application and retrieve the message file in the normal way. SEE/Change will now register the message file based on the version in the configured live or named library as specified. This processing can be observed in the form of a informational message displayed whilst the registration process is taking place, as demonstrated below:

Retrieve CR Object (RTVCROBJ)	
Type choices, press Enter.	
IR Number:	
Bottom F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display	
F24=More keys Loading messages from ACMELIVOBJ/ACMEMSGF.	

 Once complete the CR may be discarded. However, promoting the CR to live would represent best practice from an audit point of view and ensure message file consistency across all environments and machines.

5.30.4 Further message file notes

The application message file library may be shared by more than one application. In the case that two or more applications share the same application message file library and also have a message file of the same name, then one of the applications shall own the message file. The application that owns the message file shall be the first that was registered. It is assumed that library lists or delivery mechanisms (i.e. overrides) are organised so that the sharing applications' run time environments use the message file as delivered by the owning application.

If the message files are distinct then different message file libraries should be assigned for each application.

5.31 Cross Referencing take on aid (4.4304)

Briefly, the procedure to establish cross referencing for a given application is to run the take on job and review any errors. This process is iterated until all the errors are eliminated. Once complete, the application may be switched from *DATA to *FULL mode.

In this release the Cross Referencing take on job, Refresh Cross Reference Data (REFXREF), is now accompanied by a new function to aid in diagnosis of errors.

This new function is accessible via the Work with Application Take on Jobs (WRKAPPTOJ) panel as follows:

Change Management System Application Take on Job Console											
Application : ACM ACME Order Processing											
Type options, press Enter. 1=Run Job 5=Work with Result 8=Work with Last Job											
Opt Take on Job (run in sequence listed) Last Run Status											
Build Source Register	(UPDSRCREG)	11/10/04	14:59	Run/OK							
Build ILE reference information	(BLDILEREF)	00/00/00	00:00	Run/OK							
Build Non-source based objects register	(UPDNSOREG)	00/00/00	00:00	Run/OK							
5 Build Cross-Reference database	(REFXREF)	00/00/00	00:00	Run/Err							
				Bottom							
F1=Help F3=Exit F5=Refresh F9=Command F1	2=Previous										

When selected, the new Work with Cross Reference Refresh enquiry (WRKREFXREF) is presented.

Change Management System Cross Reference Refresh Enquiry								
Application : ACM ACME Order Processing System Application Status . : Run/Err Library : *ALL								
Type options, press Enter. 5=Work with Detail								
Opt Status Description *OK Pre-compile successful *W02 Non-source based program found *E03 OPM Source Not found - check source pool *E04 ILE Source Not found - check source pool	Count 149 3 2 6							
	Bottom							
F1=Help F3=Exit F5=Refresh F9=Command F12=Previous F16=REFXREF								

In this example some errors have occurred and some warnings have occurred. If selected with option *5=Work with Detail* a further panel is presented allowing further investigation.

Change Management System Cross Reference Refresh Enquiry										
Application: ACMACME Order Processing SystemApp Status : Run/ErrLibrary . : *ALLStatus : *E03OPM Source Not found - check source pool										
Type options, press En 1=Run REFXREF 8=Brow	Type options, press Enter. 1=Run REFXREF 8=Browse Cross Reference Data									
Opt Object Type	Attribute	Library	Src Level	Environment						
CST0054R *PGM CST0752R *PGM	RPG RPG	ACMELIVOBJ ACMELIVOBJ	*IGN 001 *IGN 001							
Bottom F1=Help F3=Exit F5=Refresh F9=Command F12=Previous										

Note the source level and qualifier may be set to a value commencing "*IGN". This is an internal code and indicates that this program is to be ignored by cross referencing.

To resolve warning and error conditions it will be necessary to carry out investigation and take corrective action. The take on job will then require re-running to eliminate the errors and register the programs fully before they will participate in movements arising due to cross referencing.

5.32 **REFXREF** enhancements (4.4304)

In addition to addressing a number of general SPRs, the Refresh Cross Referencing (REFXREF) processing has undergone further enhancements. These are described in the following subsections.

5.32.1 New parameter to reuse cached compilation data option

The Refresh Cross Referencing (REFXREF) process performs nongenerative compilations of objects that are under inspection. A number of techniques are used to extract information from the compiler to determine a list of referenced objects. Some of this data is stored in a local cache and is reused in subsequent REFXREF processing if the current object timestamp matches that of the cache, thus avoiding unnecessary compilation processing.

Sometimes it is desirable to rerun REFXREF without using previously cached data. It is now possible to do this by specifying the new REUSEDTA parameter:

Refresh Cross Reference Data	(REFXREF)
Type choices, press Enter.	
Application APPL Object OBJECT Type TYPE Library name LIB Re-use compile time data REUSEDTA	*ALL *ALL *ALL *YES
F3=Exit F4=Prompt F5=Refresh F12=Cancel F24=More keys	Bottom F13=How to use this display

The default for the REUSEDTA parameter is *YES, meaning cached data shall be reused where possible and the REFXREF run time is reduced accordingly. If set to *NO the cache will be cleared of existing data pertaining to objects having their reference information rebuilt and thus all objects will be compiled.

5.32.2 Audit report

Previously REFXREF only generated output if an exception was encountered. At this release processing has been changed to generate a report every time it is run and provide a summary of the activity that occurred. The report has also undergone some general formatting improvements.

5.33 Functions Added to User Authorisation Panel (4.4304)

The new Update Non-source based Objects Register (UPDNSOREG) and the existing Refresh Cross Reference information (REFXREF) functions have been added to the Work with User Authorisations panel (WRKUSRAUT).

Change Management System									
	Work with User Enrolmen	nt Details			Mode: *UPI)			
User/Group Pro	file : TRENT								
			Auth	F	ilter				
Prd			Level		Allow				
(P) Function		Authorised?	(P)	Name	(P) Chg?				
	< Locate function								
CFG ADDSIT	Add Site	Y							
CFG CFGBAROPT	Configure Bar Option	Y							
CFG CHGOBJDFT	Change Object Defaults	Y							
CFG CHGSBMDFT	Change Submit Defaults	Y							
CFG CHKCFG	Check Configuration	Y							
CFG EDTEXCMSG	Edit Execution Message	Y							
CFG REFXREF	Refresh Cross Reference Data								
CFG UPDNSOREG	Update None-Soure Based Object	Register _							
CFG UPDSRCREG	Update Source Register	Y							
CFG WRKAPPCFG	Work with Application Configura	ation Y							
CFG WRKLIBSTS	Work with Library Status	Y							
CFG WRKOBJAUT	Work with Object Authorities	Y							
CFG WRKOBJAUT	Work With Object Authorities -	Browse Y							
					More				
F1=Help F3=Ex	it F4=Prompt F9=Cmd F12=Cance	el F13=Repea	at F1	5=Temp	late				
F21=Filters F	24=Messages								

After upgrading to (or beyond) this PE, users requiring access to these management functions shall require their user authorisations amending accordingly.

5.34 Enhanced Electronic Movement Authorisations (4.4304)

Electronic movement authorisations have been enhanced to offer additional control over reversion movements. Under the new regime it is now possible to require electronic signature for the following movement types:

- Revert to Development from Live/Prod
- Revert to Development from Ready for Release
- Revert to Development from Acceptance
- Revert to Development from Module/Integration

These new functions are implemented through a redesign of the existing electronic signoff panels as shown in the following sub-sections.

5.34.1 Establishing Electronic Authorisation

Electronic authorisations may be applied at system, application or CR level and are configured through the Work with Authorisation Lists function. This function is accessible in the following ways:

- From the command line for system level authorisation configuration. The command is WRKTAL;
- From within Work with Application Configuration (WRKAPPCFG option 40) for application level authorisation configuration;
- From within Work with Change Requests (WRKCHGRQS option 40) for CR level authorisation configuration.

In all cases the following redesigned panel is presented, stating the level being worked with. In the following example the ACME application level authorisations are being maintained:

Change Management System Work with Authorisation Lists										
Level : *APP ACM ACME Order Processing System										
Use any character to select one or more of authorisation. Then press Enter.	f the following for CR movement									
Movement Description	< Forward> < Revert> Select (Code) Select (Code)									
Module/Integration library	*MDL*RDM									
Acceptance/QA library	_ *ACP _ *RDA									
Ready for Release	_ *RDY _ *RDR									
Allocate CR to a release	*ALC									
Create Release Packet	*RLS									
Live/Prod library	_ *LIV _ *RDL									
F1=Help F3=Exit F9=Cmd F12=Cancel										

Levels having authorisation lists configured for them are shown in red. The above example shows that electronic authorisation lists have been configured for:

- forward movement to the Acceptance environment;
- revert movement from Ready for Release status;
- revert movement from Live/Prod status.

Because a revert movement always reverts the CR to development, intermediate authorisations must also be met before the movement can be carried out.

For example, lets suppose the authorisations have been established:

- revert from Ready for Release requires ELAINE's signature;
- revert from Live/Prod requires DOMINIC's signature.

To revert a CR from Ready/Rls would require only ELAINE's signature, whereas to revert the same CR from Live/Prod would require both ELAINE's and DOMINIC's signature.

5.34.2 Granting and Revoking Movement Authorisations

Previously, options 41=Grant Movement Authorisation and 42=Revoke Movement Authorisation from the Work with Change Requests (WRKCHGRQS) panel presented a pop-up window. Here, the user could grant or revoke their authority for a particular movement type. These pop-ups have been replaced by a full screen panel. The new panel is designed to complement the configuration panel redesign and incorporates the new authorisation types. The following example shows the panel presented when option *41=Grant Movement Authorisation* is taken against a CR.

Change Management System Grant Movement Authorisation									
Application : ACM ACME Order Processing System CR : 001748 / 03 Customer Unit Pricing Promotion Status : *DEV - Developmnt: Under Development 28/10/04									
Select one or more movement type(s).									
Movement Description	Select	(Code)	Select	(Code)					
Module/Integration library	_	*MDL	_	*RDM					
Acceptance/QA library	_	*ACP	_	*RDA					
Ready for Release	_	*RDY	_	*RDR					
Allocate CR to a release	_	*ALC							
Create Release Packet	_	*RLS							
Live/Prod library	_	*LIV	_	*RDL					
F1=Help F3=Exit F9=Cmd F12=Cancel									

5.34.3 Working with Movement Authorisations

The Work with Movement Authorisations (WRKMVTAUT) panel, accessible from the Change Manager menu option 21, has also been revised to incorporate the new authorisation types:

	Change Management System Work with Movement Authorisation																
	User Profile : PAUL Application : ACM Group Profile: OPGMR																
Use "X" to grant, or blank to revoke movement authorisation for each of the following CRs. Then press Enter. User Authorisation																	
	IR/CR	Text				Sts M	MDL	ACP	- RDY	ALC	- Mo RLS	veme LIV	nt I RDL	'ype RDR	RDA	RDM	
	001748 03 001748 02 001748 01	Customer Customer Customer	Unit Unit Unit	ACME ACME ACME	Order Order Order	*DEV *DEV *DEV							-	-			
	F1=Help 1 F11=Chg to	F3=Exit F o group pr	5=Refr	esh autho	F7=Pre	evious .on H	s Ap F12=	p E Canc	78=Ne	ext A F24=	App -Mess	F9=C sage	omma	ind			

5.34.4 Unauthorised movements

When a user attempts to perform a movement that is unauthorised, they are presented with a series of panels that detail the authorisations required. The design of these has not changed in this PE other than to incorporate the new authorisation types as shown in the following example:



Selecting the requirement movement authorisation reveals the users that may satisfy the authorisation:

Work with Authorisation Requests							
IR/CR Number .: 000268/01 Level : *APP Application Level Authorities Movement Type .: *RDL Revert to Development from Live							
The following Users 1	nave not yet authoris	ed this Movement.					
	Authorisation List 1 of 1						
User RICHARD	User	User	User				
F1=Help F3=Exit 1 F16=Send authorisatio	59=Cmd F12=Cancel on request messages						

5.34.5 Electronic Movement Authorisation by Message

In order to gain authorisation to perform the movement electronically the user may use the F16=Sent authorisation request messages function.

This functionality has not been changed in this release.

5.34.6 Migration from earlier releases.

In PE 4.4202 a reversion authorisation type of *RDV was introduced. This allowed users to require an electronic signoff for revert movements from live. This authorisation type is replaced with *RDL under the new regime. Accordingly upgrading to PE 4.4304 will migrate *RDV authorisation requirements to the new *RDL type.

5.35 Updating Object Source Details for Delivered Objects (4.4304)

When SEE/Change delivers a source based object, its service attributes will be stamped with the source file and source file library from where it was originally compiled. This will not correspond with the real location of the source after it has been stored in the configured source pool.

In this PE a new parameter has been introduced to enable this link to be correctly maintained after objects, and their source, have been delivered. The new parameter is Update Object's Source File Details (@USD) and is maintained through the Work with Parameter Data (WRKPRMDTA) function.

Change Management System Work with Parameter Data									
2=Change 5=Display									
Opt	Code	Description	 VAR	Para LEN	DEC	er a DSP	ttril EDT	VLD	s AUT
	@STL	SEE/Change Panel Style	-	1	A			R	2
	@SVD	Libraries for THENON daily save (SAVE)	10	40	A				2
	@SVL	THENON Save Library		10	A				2
	@ svw	Libraries for THENON weekly save (SAVE)	10	40	A				2
_	@SYN	Synon/2 release level		3	A			R	2
	@ TXO	Report Heading		40	A				2
2	@USD	Update Object's Source File Details		4	A				2
_	@WRD	Word processing facility		4	A			V	2
_	@WRP	Work Library Prefix		2	А				2
_	CRSQ	CR Movement Queuing Mnemonics	4	40	A				2
_	CRSX	CR Movement Execution Mnemonics	4	40	A				2
_	CRTP	CR Type Mnemonics	4	40	A				2
_	DCAT	Default IR Category		10	A				2
_	ICAT	IR Category	10	40	A				2
	IPTY	IS Priorities (for Change Requests)	1	40	A				2
								Mo	re
F1=1	Help	F3=Exit F6=Create F9=Cmd F11=Delete	F12=0	Cance	el :	F21=	Print	t li:	st

Change Management System Work with Parameter Data							
Attr: *LEN: 0 / 4 A *DSP: *STD *EDT: *STD	*VLD: *STD *VL1: *VL2:						
Enter/update parameter v	alue: @USD : Update Object's Source File	Details					
	Parameter Value Upd Obj Details *YES	_					
		Bottom					
F1=Help F3=Exit F9=Cmd	F12=Cancel	Boccom					

When set to *YES and performing a movement to live, SEE/Change sets the source file and source file library attributes for all delivered objects having a source usage of *COMPILE, to that of the configured source file and source file library.

5.36 Release Forwarding (4.4305)

5.36.1 Introdiction

SEE/Change has been enhanced to support the onward forwarding of releases from a given production system to other production system(s). For example, the following configuration is possible:



There is no limit to the number of systems that may participate in a particular network configuration. It is possible to distribute from each of the nodes to many others, each of which may or may not also be responsible for onward forwarding. Here's another example:



In the above example, SY1 and SY2 are referred to as *Intermediate Systems*. Under the new arrangements it is possible to have an intermediate system initiate release forwarding manually or automatically. For instance, in the above example forwarding from SY1 to SY2 and SY3 is initiated manually, whereas forwarding from SY2 to SY4 is automatic.

The rest of this section describes how to configure and use the new release forwarding functionality. The above model is used as a working example.

5.36.2 Configuring Release Forwarding

Before Release Forwarding can be used, a number of configuration steps are necessary. These are described below.

5.36.2.1 Pre-requisite

It is a prerequisite to configuring release forwarding that a consistent configuration is present across the entire network. The DSTCFG and INSTALCFG utilities can be used to help achieve this, however, use of this utility requires a consistency of some library names across the network of machines. It is necessary that the entire network be upgraded to this PE level to use this new functionality. The DSTCFG / INSTALCFG commands have been changed in 4.4305 and are not compatible with earlier releases of SEE/Change. If being used on a network that has only been partially upgraded, contact your vendor support department for advise.

5.36.2.2 Configuring the Release Packet Archive

When an intermediate system receives a release packet, it is necessary to store it so that it is available for onward forwarding. These release packets are held in a nominated library known as the *Release Packet Archive*. The release packet library is configured through the introduction of a new general parameter, as shown below:

Change Management System							
Work with Parameter Data							
2=0	Change	e 5=Display					
				Para	ameter a	attribut	es
Opt	Code	Description	VAR	LEN	DEC DSI	P EDT VL	D AUT
					_		
_	@RFC	Release Forwarding DEVCTR Status Checks		4	A	V	2
	@ROV	Allow overrides for DMS submit RCVRLS?		4	A	V	2
2	@RPA	Release Packet Archive Library		10	A		2
	@SBM	Job Description for submitted CR jobs		10	A	V	2
_	@SDL	Suffix for restore of DMS library		2	A	R	. 2
_	@SHL	Shipment Libraries	10	40	A		2
-	@SNH	Outgoing data transfer hold status		4	A	Y	2
_	@SNL	Send to Live from WRKCHGRQS		4	A	Y	2
-	@SNT	Send Type default TAPE/COMS		4	A		2
-	@SSJ	Site Specific Job Descriptions 4 Compile		4	A	V	2
_	@STL	SEE/Change Panel Style		1	A	R	. 2
_	@SVD	Libraries for THENON daily save (SAVE)	10	40	A		2
-	@SVL	THENON Save Library		10	A		2
_	@svw	Libraries for THENON weekly save (SAVE)	10	40	A		2
_	@SYN	Synon/2 release level		3	A	R	2
						М	ore
F1=H	Help	F3=Exit F6=Create F9=Cmd F11=Delete	F12=0	Cance	el F21=	=Print l	ist

Modify the parameter value to specify your release packet archive library.

Change Management System Work with Parameter Data							
Attr: *LEN: 0 / 10 A *DSP: *STD *EDT: *STD	*VLD: *STD *VL1: *VL2:						
Enter/update parameter v	alue: @RPA : Release Packet Archive Library						
	Parameter Value Library Name @RPALIB						
		Bottom					
F1=Help F3=Exit F9=Cmd	F12=Cancel						

If unspecified, or set to *NONE, the local system will not behave as an intermediate system, irrespective of other settings.

Release packets are stored inside the release packet archive library in save files. The save files are named in the following format:

dddRLnnnnn

where *ddd* represents the originating development system code and *nnnnn* represents the release number. Because this format does not conflict with any other naming format used elsewhere in SEE/Change, it is possible to use an existing archive library for this purpose in addition to its existing purpose.

The Release Packet Archive parameter must be set up on all systems that are to be intermediate systems. The @RPA parameter is distributed and installed by the DSTCFG / INSTALCFG function. Therefore, if you use DSTCFG and use the same library name on all your systems, you may configure this just once and distribute your configuration.

5.36.2.3 Configuring Application Distribution

It is necessary to tell SEE/Change the distribution pattern that is to be used across the network. This is done, by application, through the introduction of a new option within Work with Application Configuration:

THNSPT Change Management System Work with Application Configuration							
18=IFS Paths 30=Take On Jobs 35=Distribution 40=Wrk auth 1st 43=IR dft schd 44=CR dft schd							
Opt App Description 35 ACM ACME Sales Order Processing Bottom							
F1=Help F3=Exit F4=Prompt F5=Refresh F6=Create F9=Cmd F12=Cancel F21=Include Thenon app F23=More options F24=Messages							

It is also accessible via the native command interface. The command is WRKAPPDST and accepts an application code as a parameter:

WRKAPPDST APPL(xxx)

When selected, a display like the following is presented:
Change Management System Work with Application Configuration - Distribution	
Application : ACM Acme Sales Order Processing Local	System : DEV
Type options, press Enter. 7=Select Systems to Distribute To 9=Toggle Auto-Forwarding	
Opt System / Description TRD TRAININGD - Development Centre SY1 Test System SY2 Validation System SY3 Training System SY4 Live System	Auto- Level Fwd to? 0 1 1 1 1
F1=Help F3=Exit F5=Refresh F9=Command F12=Cancel F19=Left	Bottom F20=Right

By default all distribution is assumed to be from the development centre and is denoted by the indentation as shown above. Note also the level indicator, which may be useful where the depth of the distribution model exceeds the limitations of the display width.

In order to achieve our distribution model it shall be necessary to tell SEE/Change to distribute from SY1 to SY2 and SY3. By implication this shall define SY1 as an intermediate system. To do this select option 7=Select Systems to Distribute To against SY1. The following display is presented:

Change Management System			
Work with Application Config	uration - Distribution		
Application : ACM Acme Sales Orde: Distribution from : SY1 Test System	r Processing Local System : DEV		
Select systems to distribute to.			
Sel System / Description 1 SY2 Validation System 1 SY3 Training System SY4 Live System	Current distribution from TRD TRAININGD - Development Centre TRD TRAININGD - Development Centre TRD TRAININGD - Development Centre		
F1=Help F3=Exit F5=Refresh F9=Command	Bottom F12=Cancel		

Here, we are presented with a list of the systems that are eligible for distribution from SY1. Note the list excludes the development centre and the selected system.

We select SY2 and SY3 as per our desired configuration and press <Enter> to continue. The main configuration panel is redrawn as shown below:

SEE/Change – Supplementary Manual 4.5003

Change Management System Work with Application Configuration - Distribution		
Application : ACM Acme Sales Order Processing Local	System	: DEV
Type options, press Enter. 7=Select Systems to Distribute To 9=Toggle Auto-Forwarding		Auto-
Opt System / Description TRD TRAININGD - Development Centre SY1 Test System SY2 Validation System SY3 Training System SY4 Live System	Level 0 1 2 2 1	Fwd to?
F1=Help F3=Exit F5=Refresh F9=Command F12=Cancel F19=Left B	F20=Righ	Bottom nt

Next, we wish to set SY2 as an intermediate system distributing to SY4. Executing option 7 against SY2 will once again present the system selection:

Change Management System Work with Application Configuration - Distribution			
Application : ACM Acme Sales Order Distribution from : SY2 Validation Syste	r Processing Local System : DEV em		
Select systems to distribute to.			
Sel System / Description SY3 Training System SY4 Live System	Current distribution from SY1 Test System TRD TRAININGD - Development Centre		
F1=Help F3=Exit F5=Refresh F9=Command	F12=Cancel Bottom		

Note that SY1 does not appear in the list, since it is responsible for distribution to SY2. It is possible, however, to reconfigure the distribution for SY3 to be via SY2, instead of via SY1. In our case we simply want to select SY4. After selection, the main configuration panel is redrawn once again:

SEE/Change – Supplementary Manual 4.5003

Change Management System Work with Application Configuration - Distribution			
Application : ACM Acme Sales Order Processing Local	System : DEV		
Type options, press Enter. 7=Select Systems to Distribute To 9=Toggle Auto-Forwarding			
Opt System / Description TRD TRAININGD - Development Centre SY1 Test System SY2 Validation System SY4 Live System SY3 Training System	Auto- Level Fwd to? 0 1 2 3 2		
F1=Help F3=Exit F5=Refresh F9=Command F12=Cancel F19=Left	Bottom F20=Right		

At this stage all onward distribution is manual. In the case of distribution from SY2 to SY4, we require the distribution to be automatic. In order configure this we need to select option *9=Toggle Auto-Forwarding* against the *target* system, SY4 in this case. Afterwards the display shows as follows:

Change Management System Work with Application Configuration - Distribution				
Application : ACM Acme Sales Order Processing Local	System : DEV			
Type options, press Enter. 7=Select Systems to Distribute To 9=Toggle Auto-Forwarding				
Opt System / Description TRD_TRAININGD - Development Centre SY1_Test System SY2_Validation System SY4_Live System SY3_Training System	Auto- Level Fwd to? 0 1 2 3 2 3			
F1=Help F3=Exit F5=Refresh F9=Command F12=Cancel F19=Left F	Bottom F20=Right			

The distribution configuration is now complete.

The distribution configuration must be set up identically on all systems.

The distribution configuration data is held in the O#RFC file and is distributed by the DSTCFG / INSTALCFG function. Note that all systems receiving configuration data via the DSTCFG / INSTALCFG function must be at the same release level. This is also a pre-requisite for the use of this new functionality.

5.36.2.4 Configuring a Development Centre Check

A release may only be forwarded from an intermediate system provided that none of the CRs contained within it are reverted at the local, intermediate system. It is further possible to prevent manual forwarding from an intermediate system if any of the CRs in the release have been reverted *at the development centre system*. This check is implemented through the introduction of a new general parameter, as shown below:

Change Management System Work with Parameter Data									
	2=0	Change	5=Display						
					Para	ameter a	ttribu	ites	s
•	Opt	Code	Description	VAR	LEN	DEC DSP	EDT V	/LD	AUT
	0	0.5.7.9			4				0
	<u> </u>	@RFC	Release Forwarding DEVCTR Status Checks		4	A		V	2
	_	@ROV	Allow overrides for DMS submit RCVRLS?		4	A		V	2
	_	@RPA	Release Packet Archive Library		10	A			2
	_	@SBM	Job Description for submitted CR jobs		10	A		V	2
	_	@SDL	Suffix for restore of DMS library		2	A		R	2
	_	@SHL	Shipment Libraries	10	40	A			2
	_	@SNH	Outgoing data transfer hold status		4	A		Y	2
	_	@SNL	Send to Live from WRKCHGRQS		4	A		Y	2
	_	@SNT	Send Type default TAPE/COMS		4	A			2
	_	@SSJ	Site Specific Job Descriptions 4 Compile		4	A		V	2
	_	@STL	SEE/Change Panel Style		1	A		R	2
	-	@SVD	Libraries for THENON daily save (SAVE)	10	40	А			2
	-	@SVL	THENON Save Library		10	А			2
	-	0 SVW	Libraries for THENON weekly save (SAVE)	10	40	A			2
	-	ASYN	Synon/2 release level	10	3	A		R	2
	-	GOIN	Synon/2 rerease rever		5	11		Mo	~
								101	.e
	F1=F	Help	F3=Exit F6=Create F9=Cmd F11=Delete	F12=0	Cance	el F21=3	Print	lis	st

When the parameter value is set to *YES the validation will take place when manually forwarding a release from an intermediate system.

Change Management System Work with Parameter Data			
Attr: *LEN: 0 / 4 A *VLD: Value List *DSP: *STD *VL1: *YES *EDT: *STD *VL2: *NO	::		
Enter/update parameter value: @RFC : Release Fo	prwarding DEVCTR Status Checks		
Parameter Value *YES or *NO <u>*</u> YES			
	Bottom		
F1=Help F3=Exit F9=Cmd F12=Cancel			

Note that this flag is only checked when manually forwarding. When the intermediate system is configured to auto forward, no validation is performed; the release is simply forwarded irrespective.

5.36.3 Purging the Release Packet Archive

After a time, the release packet archive library will require purging. A new command, Purge Release Packet Archive (PRGRPA), has been provided to perform this task:

Purge Release	Packet Archive (PRGRPA)	
Type choices, press Enter.		
System code, or *ALL	Name, *ALL Date, *CURRENT	
F3=Exit F4=Prompt F5=Refresh F24=More keys	F12=Cancel F13=How to use this display	om

The command requires two parameters; the *Development Centre System* and the *Receive Date*. For instance:

PRGRPA SYSM(DEV) ENDT(300904)

attempts to remove release packet save files from the release packet archive library for all releases received locally from the DEV development centre system on or before 30th September 2004.

Since this command is driven by data held within the Release Manager, which is subject to purging by the Purge Change Management Data (PRGCHGDTA) command, this purging command has also been modified to automatically remove the release packet if the release record is purged from the SEE/Change Database.

5.37 Enhanced ILE Remote Delivery Method (4.5002)

Previously the *ILE Remote Delivery Method* could be set to MOD or PGM and was used to influence the release packaging process in respect of ILE program. This has been extended at this release to incorporate a third option as demonstrated below:

THENON THENON Change Manageme	nt System 4.5002
Work with Applica	tion Details
Document Processing Enabled <u>N</u> Document Top Level Folder ACM	(Y)es/(N)o
ILE Processing Enabled \underline{Y}	(Y)es/(N)o
ILE Remote Delivery Method <u>*BOTH</u>	*PGMONLY/*MODONLY/*BOTH
ILE Module Pool Library <u>ACMELI</u>	VOBJ
ILE Allow missing modules Y	(Y)es/(N)o
Cross Referencing Utilisation <u>*FULL</u>	*NONE/*DATA/*FULL
JDE Application? N	(Y)es/(N)o
SEE/Change Version for JDE 00	(00 - 99)
Development Prod. System(P).	
Development Prod. Site(P).	
Unlock Source at Dev-Prod System . $\underline{\mathbb{N}}$	
IFS Processing Enabled \underline{N}	IFS Archive Library
IFS CR No. of Archive Levels $\underline{00}$	
CRTCRLIB to create schema? N	(Y)es/(N)o
Maintain existing triggers? <u>Y</u>	(Y)es/(N)o
Set file attributes from CR? \underline{Y}	(Y)es/(N)o
	Bottom
F1=Help F3=Exit F4=Prompt F9=Cmd F12=	Cancel F16=Update F24=Messages

The following table shows the pre and post upgrade possible values along with their meanings. In this table the term *program* refers to both *Program* and *Service Program* objects.

Pre 4.5002	Post 4.5002	Meaning
PGM	*BOTH	Modules <i>and</i> program objects are packaged into the release packet. When received, the existing program is <i>moved</i> to create an archive version. The distributed program is delivered to the target library and the modules rebound to it (after recompilation, if so configured).
MOD	*MODONLY	Only the modules are packaged into the release packet. Programs are NOT packaged. When received, the existing program is <i>duplicated</i> to create an archive version. The live program is then updated with the new modules (after recompilation, if so configured).
N/A	*PGMONLY	New setting introduced at 4.5002. Only program objects are packaged into the release packet. When delivered, the existing program is <i>moved</i> to create an archive version and the distributed program is then delivered to the target library. Rebind processing is bypassed.

By using the new setting release packet sizes and installation times are can be reduced. However, release packet size is not the only consideration to be borne in mind when modifying this parameter setting, so it is important to ensure that use of this new setting is appropriate for your configuration before making any changes.

5.38 Create Object Type (CRTOBJTYP) Command (4.5002)

A new utility command has been introduced to make it easier to create new user defined object types:

Cre	eate Object Type (CRT	OBJTYP)
Type choices, press Enter.		
Base Object Type Base Object Attribute New Object Type New Object Attribute New OMS Reference Name Execution Message ID		Character value Character value Character value Character value Character value Hexadecimal value
F3=Exit F4=Prompt F5=R6 F24=More keys	efresh F12=Cancel	Bottom F13=How to use this display

The command works by replicating XOT and XIF records for an existing object type and changing key values to suit. It is therefore necessary to specify the most suitable existing object type in the first two parameters (type and attribute). Specify the new object type, attribute and SEE/Change name in the following three parameters. A previously unassigned Execution Message ID must be specified in the last parameter and must be within the F0 to FF range.

A supplementary document called "Creating new object types" is available from the Thenon download site or your local software vendor. It is recommended that a copy of this document is obtained and studied before using this command.

5.39 Site Specific Configuration having no Database Library (4.5002)

Previously the Work with Application Configuration (WRKAPPCFG) function insisted that site specific configurations had to have both object and data libraries specified, implying that the application must be structured like this:



Should an application have a common database but require site specific functionality, the model would look like this:

SEE/Change – Supplementary Manual 4.5003



The Work with Application Configuration function has been changed to allow the above configuration simply by omitting the data libraries from any three of the sites. Accordingly either configuration is now possible.

Note that one site must still be specified for the distribution of the common data files. In the following screen print (which is in fact an amalgamation of two screen prints to make for easier reading) the above has been configured for the WAP application. Notice that the London site has been used to specify where the base files are to be delivered. It could have equally been any of the other three sites.

THENON THENON Change Management System 4.5002 Work with Application Configuration - Libraries			
Application	: WAP World Applica	tion Lo	cal Sys. : D43
12=Obj typ overrid	des 13=CR type override:	s 15=Environment sett	ings
Opt System Typ	pe Type:Description	Target Libra	ry Ovr
		Live/Prod Accept/QA	Integr/Tst
Developmen Dev	v Obj: Base application	WAPLOBJ WAPAOBJ	WAPMOBJ
	Obj: Site Hong Kong,	WAPLOBJHK WAPAOBJHK	WAPMOBJHK
	Obj: Site London, UK	WAPLOBJLON WAPAOBJLON	WAPMOBJLON
	Obj: Site New York, U	WAPLOBJNY WAPAOBJNY	WAPMOBJNY_
	Obj: Site Sydney, Aus	WAPLOBJSYD WAPAOBJSYD	WAPMOBJSYD
	DB : Base Development	WAPLDTA WAPADTA	WAPMDTA
Draduction Dra	d Obi. Dece application		
PIOduction Pic	Obj: Base application	WAPLOBJ WAPAOBJ	
	Obj: Site London UK	WAPLOBJIK WAPLOBJIK	
	Obj. Site New York U	WAPLOBINY WAPLOBINY	
	Obj: Site Sydney, Aus	WAPLOBISYD WAPLOBISYD	
	DB : Base Hong Kong.		
	DB : Base London, UK	WAPLDTA WAPADTA	
	DB : Base New York, U		
	DB : Base Sydney, Aus		
			Bottom
F1=Help F3=Exit	F5=Refresh F9=Cmd F12=	=Cancel F16=Update F	24=Messages

5.40 Cross Referencing used with User Defined Object Types (4.5002)

Cross Referencing has been enhanced to support relationships between objects that are of user defined types as well as Thenon predefined object types.

5.41 Stream File Support (4.5003)

5.41.1 Introduction

At release 4.5003 SEE/Change's support for stream files has been significantly enhanced over the earlier '*IFS Support*' introduced in 1999/2000 with version V4.4200. To simplify the discussion, the earlier system is referred to as *IFS support*, and the newer as *stream file support*.

Stream file support brings both new functionality and new terminology. However the major stimulus for its introduction has been the development of a SEE/Change Eclipse plug-in for IBM's WebSphere Development Studio Client for iSeries, subsequently referred to as the *WDSc*. This plug-in reproduces most of the 5250 SEE/Change Development Manager's functionality for library objects, but also contains new stream file features that could not be included in the 5250 Development Manager. These capabilities make it possible to exert change management (CM) control over 'mixed' applications, that is applications consisting of both OS/400 library objects and stream files objects containing, say, HTML or Java source or binaries.

Thenon therefore recommend that users evaluating stream file support do so via the WDSC client interface. Even if WDSC is not your preferred IDE (e.g. your mixed application contains library objects and VB objects) you will probably find it easier to manage your stream files in this way.

5.41.2 Overview

The diagram below and the footnotes that follow are intended to give an end-to-end overview of stream file support:



The following terms are noteworthy:

- Registration: Initial registration of a folder and its content in readiness for change management. During registration an image of the folder content is recorded in the application's *Folder Register*.
- Folder Register (or register): SEE/Change's image of the folder's files and subdirectory structure that is currently live. The folder register is created by the registration process and thereafter updated by movements to and from the live environment.
- Retrieval: Creation of a local copy of a registered folder (or part thereof) in a CR specific work folder. The CR work folder is constructed from the folder register. The same folder can be retrieved into multiple CR's at the same time in order to construct a complete environment in which an IDE can operate; only files that change are considered for onward promotion.

- Resolve Conflicts: Conflicts can occur for a number of reasons. These are discussed in more detail in the following sections. All conflicts must be resolved before the CR can be moved out of development.
- Check-in: Once all conflicts are resolved the CR may be moved to Module/Integration, Acceptance or Live. Check-in occurs when the first movement takes place. At this time a list of all changed (new, changed or deleted) stream files is calculated and images of these are recorded into a CR specific area (checked in). The CR work folder is then saved and removed from the IFS to prevent further invalid changes taking place.
- Register Update: When the CR is moved to live, the folder register is updated with the new checked-in version of the changed file(s);
- Folder Update: The movement mechanism delivers the new version of changed files to the configured target folder. If this folder happens to be in the QNTC file system, then it is delivered directly to the remote server subject to that server's availability. If the file is replacing an existing version and archiving is switched on, then a copy of the version being replaced is made in the *folder archive*. Archiving is analogous to that of library object support.
- *Folder Archive (or Archive)*: A repository of previous versions of replaced (or deleted) stream files. The archive is used when live CRs are reverted.

Sections 5.41.3 to 7.17.1 describe how to enable and use Stream File Support whilst section 5.41.5 provides further more detailed information on the topic of handling stream files.

5.41.3 Folders

SEE/Change provides CM control over *folders* that are arbitrary directories of stream file objects. In general folders are subdirectories that can contain both stream files and other subdirectories. The term *folder* replaces *path* as used in IFS support. For registration, development, testing and distribution purposes folders can reside anywhere in the root part of the IFS of the SEE/Change development system, or on any NT server that is visible in the QNTC file system of the IFS. At the end of the CM cycle changed stream files are archived in physical files in an application-configured OS/400 library, so that SEE/Change acts as a secure repository for all objects in the mixed application.

5.41.4 Folder Registration

To allow a folder to participate in CM operations, it must first be registered to a SEE/Change application. Folder registration is a function of the Configuration Manager. During registration you provide the location of the folder in either the root file system or in shares on an NT server visible via the QNTC file system. This location is analogous to the libraries named as the live (*LIV) environment for the application's library objects. The registered folder may be empty, or may contain stream files and subdirectories. If the folder is not empty, it is scanned and an initial registration copy of the folder contents is taken at this time. This is stored in SEE/Change's database and is referred to as the *register*. When scanning the folder, certain files can be ignored as determined by the setting of the @/GN general parameter. This parameter contains a list of regular expressions (e.g. *.tmp, *.bak) that are matched with filenames encountered during the scan.

You can also specify intermediate target file system locations that correspond to the module/Integration (*MDL) and / or the acceptance (*ACP) testing environments. Finally, you may specify a flag setting that determines whether or not a folder should be placed in a release packet for distribution to remote iSeries systems.

For demonstration purposes we shall assume that ACME, a fictitious ebusiness company that uses mixed mode iSeries and NT application servers, has the following application architecture:



We can see from this diagram that ACME uses IFS folders for the various CM lifecycle environments on the development system, but has a dedicated NT server for testing purposes linked to the acceptance test environment on the production machine, as well as a dedicated NT server for the Live/Production environment.

5.41.4.1 Activating Stream File Support

Before registering a new folder, it is necessary to activate Stream File Support for your application and decide where your application's folder archive is to be located. To do this access the Work with Application Configuration (WRKAPPCFG) panel:

ACMEDEV ACME Change Management System Work with Application Configuration				
2=Change	3=Copy	4=Delete	5=Display	6=ASSET Cfg
9=Overrides	12=Where used	14=Src files	16=CASE info	17=JDE info
Opt App Desc: < 1 2 ACM	ription Locate Order Processing	System		Bottom
F1=Help F3=E:	kit F4=Prompt F5	5=Refresh F6=Cr	eate F9=Cmd F1	2=Cancel
F21=Include Th	nenon app F23=Mon	re options F24=1	Messages	

Select your application with option 2=*Change* and on the resulting panel press Page Down to view the second page of Application Details:

ACMEDEV ACME Change Manag Work with Applica	ement System tion Details
ILE processing enabled Y ILE Remote Delivery Method *BOTH ILE Module Pool Library ACMELT ILE Allow missing modules Y Cross Referencing Utilisation	<pre>(Y)es/(N)o *PGMONLY/*MODONLY/*BOTH VOBJ (Y)es/(N)o *NONE/*DATA/*FULL (Y)es/(N)o (00 - 99)</pre>
Stream file processing enabled . \underline{Y} CRTCRLIB to create schema? \underline{N} Maintain existing triggers? \underline{Y} Set file attributes from CR? \underline{Y}	Archive Library <u>ACMEARCLIB</u> (Y)es/(N)o (Y)es/(N)o (Y)es/(N)o
	Bottom
F1=Help F3=Exit F4=Prompt F9=Cmd F12=	Cancel F16=Update F24=Messages

Select 'Y' for the "Stream file processing enabled" parameter and specify a valid library name for the Archive Library parameter. This is the library in which SEE/Change will initially register a copy of your application's folders (the register) and into which later versions of modified stream files will be held (the archive). In fact both the register and the archive are held in a single file called O#IARC which SEE/Change will automatically create for you in the specified library. If you use an archive library for library objects, it is recommended that you specify the same library name for your stream file register / archive as well. Once you have completed the parameters, press *F16=Update* to register the changes and return to the main *Work with Application Configuration* display.

If you are distributing stream files via a SEE/Change production license, it will be necessary to enable stream file support on that machine as well. Follow the above procedure for each production machine that you wish to distribute stream files to.

5.41.4.2 Registering a new Folder

We are now ready to register a folder:

ACMEDEV ACME Change Management System Work with Application Configuration			
18=Folde 44=CR df	30=Take On Jobs 35=Distribution 40=Wrk auth 1st 43=IR df	t schd	
Opt App 18 ACM	Description < Locate ACME Order Processing System		
	1	Bottom	
F1=Help F21=Incl	F3=Exit F4=Prompt F5=Refresh F6=Create F9=Cmd F12=Cancel		

Here, *F23=More options* reveals option *18=Folders*, formerly *IFS paths*. (It is not necessary to press F23 for the option to be valid). Select the option against your application to access the Registered Folders display:



Initially no folders will be registered. To register a new folder press *F6=Register Folder*. The Register Folder display will be shown:

ACME Change Management System Work with Application Configuration - Register Folder				
Type the Folder to be registered and, optionally, an Acceptance and/or Module/Integration Folder. Press Enter. Live Folder to Register				
Acceptance Folder *NONE				
Module/Integration Folder *NONE				
F3=Exit F4=Prompt F9=Command F12=Cancel				

Here we specify the folder that corresponds to the live environment and optionally, folders that correspond to the acceptance test and module/integration test environments. The Universal Naming Convention (UNC) is used to specify folders and, for reference, this is as follows:

```
//Server/Share Name/Sub dir1/Sub dir2 etc
```

In SEE/Change, if the server name is equal to the local system name, then the share name that follows refers to a root IFS directory location, otherwise it is assumed to be the name of a server accessible via the iSeries' QNTC file system. Internally, SEE/Change addresses these by prefixing the QNTC directory to arrive at a fully qualified iSeries IFS name, i.e.

//QNTC/Server/Share_Name/sub_dir1/sub_dir2 etc

Root IFS folders do not require a network share in order to be change managed by SEE/Change.

Folder names may be entered directly else F4=Prompt may be used to scan dynamically and allow an existing folder to be located and selected. If the prompt is used a display like the one below is presented:

ACME Change Management System Work with Application Configuration - Register Folder				
: Select Dire	ectory		• • • • • • • • • • • • • • • • • • • •	:
: Path : Type option, press Enter. 1=Select 5=Expand directory Opt Directory/File //ACME_DEV //ACME_NTTST/www //ACME_NTLIV/www	Type S S S	Size(KB)	Amended 00/00/00 00:00:00 00/00/00 00:00:00 00/00/00 00:00:00	
: : : F3=Exit F12=Cancel F22=Full Path : : : : : : : : : : : : : : : : : : :	Cancel		Bottom.	: : :

In the above ACMEDEV is the name of the local iSeries server and represents the local IFS. ACME_NTTST and ACME_NTLIV are names of NT servers that are advertising network shares that are accessible via the iSeries' QNTC file system.

Because we wish to register a local IFS subdirectory as our live folder, we select option 5 to expand ACMEDEV and then, from the resulting display, select our folder, www_liv. After selecting this and the corresponding folders for the module/integration and acceptance test environments, our display will appear as follows:

ACME Change Management System Work with Application Configuration - Register Folder	
Type the Folder to be registered and, optionally, an Acceptance and/or Module/Integration Folder. Press Enter. Live Folder to Register //ACMEDEV/www_liv	
Acceptance Folder //ACMEDEV/www acp	
Module/Integration Folder //ACMEDEV/www_mdl	
F3=Exit F4=Prompt F9=Command F12=Cancel	

SEE/Change – Supplementary Manual 4.5003

Press <Enter> to continue with the registration process. You are required to provide a brief description for your folder and specify whether it is to be distributed or not. Because our folder is to be distributed, we specify (Y)es. Specifying (N)o prevents this folder from being distributed to production systems:

ACME Change Management System Work with Application Configuration - Register Folder			
Ty . Mo :	Confirm Registration of Folder	• :	
Li :		:	
:	: Application: ACM ACME Order Processing System :	:	
	Type Folder text and press Enter to confirm.	:	
Ac : // : :	Folder to Register //THENON/www_liv	:	
; Mo :: // ::	Folder text ACME e-commerce web site Distribute? $\underline{\underline{Y}}$ (Y/N)	: : :	
	: F12=Cancel	:	
F3=E	Exit F4=Prompt F9=Command F12=Cancel		

Finally, job REGFLDR is run to register the folder. The folder content, if there is any, is saved into SEE/Change's register and once complete the folder becomes available for change management.

Note that when registering folders it is the user who is running the job whose authority is used to perform the folder scans and reads.

A user having insufficient authority may therefore find that their registration fails or completes normally but is incomplete. This is because SEE/Change is dependant on the behaviour of the file system in which the folder being registered resides. This could be a folder held locally within iSeries' IFS, else on an external server accessible via the QNTC file system. For instance, if a file or subdirectory is visible, but inaccessible, registration is likely to result in error, whereas a file or subdirectory that is invisible is more likely to result in registration completing normally, but being incomplete.

It is recommended that a special profile be used for folder registration purposes to avoid errors or incomplete registration. If registering folders on the iSeries' IFS, QSECOFR or an equivalent profile should be used. If registering a folder on an external server, a profile that also exists on the remote server and has sufficient access to that folder should be used.

5.41.4.3 Configuring Production Folder Overrides

If a registered folder is set to distribute to a production system, SEE/Change assumes the target folders to be the same on the production system as the development centre. This is convenient if we intend to change manage the content of IFS folders as it is likely the folder names will be the same on each machine. For example:

	Development	Production
Module/Integration	//root/website/module/www	
Acceptance	//root/website/acceptance/www	//root/website/acceptance/www
Live	//root/website/live/www	//root/website/live/www

However, if the folder names are different, or are located on servers external to the iSeries, then it will be necessary to apply *Production Folder Overrides*.

Production folder overrides can be configured on either the development system or locally on the production system.

If configured at the development system, they will be incorporated into the release packet and applied to the production system when the release is received overwriting those previously configured at the production system, if any. That is, production folder overrides configured at the development centre take precedence over those configured at the production system.

In ACME's architecture, overrides need to be set up as follows:

	Development	Production overrides
Module/Integration	//ACMEDEV/www_mdl	
Acceptance	//ACMEDEV/www_acp	//ACME_NTTST/www
Live	//ACMEDEV/www_liv	//ACME_NTLIV/www

Let's suppose that we centrally configure these at the development centre. To do this we select option 7=*Production Folder Overrides* from the *Registered Folders* display:



The resulting display lists the systems to which the application is configured for distribution:



Option 2=Edit Override allows production folder overrides to be specified. If the override is to be specified locally at the production machine, or the target folder is to be the same as that specified for the development system, the field can be left blank.

ACME Change Management System		
Work with Application Configuration - Edit Override		
System : PRD Production		
Type Folder Name(s) and press Enter.		
Default Live Folder		
"root"/www liv		
Override to Production Live Folder		
//ACME NTLIV/www		
Default Acceptance Folder		
"root"/www.acp		
Override to Production Acceptance Folder		
//ACME NTTST/www		
F3=Exit F9=Command F12=Cancel		

Enter saves the information and returns to the previous display.

Note that on this display, the override folder is left blank, then no override will be in force. This means that the override that is set up on the production machine will be used if there is one, else the default value will be used. In the above example the default live folder is

```
"root"/www liv
```

The special value *"root"* refers to the root of the IFS on the production system.

5.41.5 Further Information on Stream File Processing

This section contains further supplemental information about how SEE/Change handles stream files.

5.41.5.1 Conversion of IFS Support to Stream File Support

If you use IFS Support the following points are noteworthy prior to upgrading.

- Existing archives and movement histories are preserved;
- Before upgrading to 4.5003 existing CR's that contain stream file objects must be promoted to live or reverted to development;
- CR's that were promoted to live before the upgrade may be reverted but cannot be re-promoted;
- CR's that are at development before the upgrade (or get reverted after the upgrade) cannot be promoted. This is because of changes to the structure of the CR work folder. Instead, a new CR must be created, the folder retrieved

into it and changed parts from the old CR work folder manually transferred into the new CR work folder, observing the directory structure accordingly.

If you used environment variables to support intermediate environments prior to upgrading (see PE Notes for release 4.4305), these can be removed after the upgrade. However, these are *not* inspected / converted by the upgrade routine and so intermediate environment folders will need to be configured into your application configuration after upgrading.

5.41.5.2 Ignored File Types (The @IGN General Parameter)

In IFS support it was necessary to explicitly state the file types that were to participate in change management. In stream file support all files are assumed to participate unless their name matches one of the formats specified in the @IGN general parameter.

By default, @IGN is shipped with the following values:

A	CME Change Management System	
	Work with Parameter Data	
Attr: *LEN: 10 / 40 A	*VLD: *STD	
*DSP: *STD	*VL1:	
*EDT: *STD	*VL2:	
Enter/update parameter ta	able: @IGN : Stream file name patterns to	be ignored
	•	2
Parameter Var	Parameter Value	
Identifier	Name Pattern	
001	*.bak	
002	*.tmp	
003	*	
004	#*	
005	~*	
		More
F1=Help F3=Exit F9=Cmd	F12=Cancel	

The @IGN parameter is applied at folder registration time and during check-in.

The @IGN values are subject to local translation and as such its content should be checked post upgrade for relevance on the local system, especially when operating in an environment using CCSIDs other than 37 and 285 in which the tilde (~) character in particular is affected.

5.41.5.3 Intermediate Environments

If intermediate target locations corresponding to the module/integration test and acceptance test environments are to be specified SEE/Change will assume that, at the outset, the content of all the specified locations is identical. If this is not the case, it will be necessary to make it so before allowing CR movements for your newly registered folder.

When registering a new folder, SEE/Change inspects only the configured live location.

5.41.5.4 Refresh Register

It might be necessary to rerun the registration process; for example, if changes have been made directly to the target environment outside of SEE/Change. In this case a Refresh Register option is provided in the Work with Application Configuration – Registered Folders panel:



As with initial registration, this process inspects only the configured live folder. If intermediate environments are used and changes have been made outside of SEE/Change, it will be necessary to ensure that the same changes are made to all environments in order to ensure ongoing environment integrity.

5.41.5.5 Sizing Considerations (Archival Overhead)

This section discusses the folder register and archive and is intended to give general guidance for sizing exercises as well as some general background information on how the register and archive file work.

When a folder is registered for the first time, an image of the folder's content is taken and stored in the application's archive library in file O#IARC. This image is referred to as the register. To give an indication of the overhead of registration the following test was carried out:

Folder Size:	602 MB
Number of sub folders:	23
Number of files:	307

The folder was registered to an application whose O#IARC file was previously empty. Afterwards, the O#IARC archive file was 618 MB in size, indicating an archive overhead of approximately 2.65% calculated as follows:

((618 - 602) / 602) x 100 = 2.65%

As with all such sizing exercises, a number of factors will influence the size of this file and the actual overhead. For instance, a folder containing a large number of very small files (say, 1 or 2 KB each) will result in a higher overhead than a folder containing fewer files of a larger size each. This is because the archive file works in 8 KB elements.

Also, the archive will increase in size according to the number of archiving levels that the application is configured to hold along with the number of changes made to the folder. To demonstrate this, let's assume that our application archiving levels is set to 2, and our entire folder is to be repeatedly checked out, every file changed in some minor way so as to register a change, and then promoted to live. The register would remain at 618 MB whilst the archive would rise to 1854 MB (3 x 618 MB). After this time it would stay the same. This would come about as follows:

 Registration; version 1 of each file is recorded in the register. O#IARC is now size 618 MB and will appear as follows:

O#IARC		
Register	Archive	
V001		

 Change number 1 will create version 2 of each file. When version 2 is promoted live, a copy is taken from the current live environment and stored in the archive as version 1. Version 2 is then delivered to the live environment and also written out to the register, replacing the previous version. O#IARC now 1236 MB and will appear as follows:

O#I/	ARC
Register	Archive
V002	V001

 Change number 2 will create version 3 of each file. When version 3 is promoted live, a copy is taken from the current live environment and stored in the archive as version 2. Version 3 is then delivered to the live environment and also written out to the register, replacing the previous version. O#IARC now 1854 MB and will appear as follows:

O#IARC		
Register	Arch	nive
V003	V001	V002

Change number 3 will create version 4 of each file. When version 4 is promoted live, a copy is taken from the current live environment and stored in the archive as version 3. Because there are now more than 2 archiving levels, the oldest version is overwritten, thus the version 3 image replaces the version 1 image. Version 4 is then delivered to the live environment and also written out to the register, replacing the previous version. O#IARC has remained at 1854 MB and will appear as follows:



The above example is extreme because it is unlikely that every file would be changed every time it was retrieved. In practice it is only files that are *retrieved and changed* that cause an archive level to be incremented or overwritten.

It is important to note that the archive is formed from the actual live file.

5.41.5.6 Performance Considerations

This section discusses the performance of stream file registration and movements.

In order to provide some indicative statistics a copy of the 602 MB folder discussed in the previous section was established in the IFS in triplicate (module/integration, acceptance and live). The folder was then registered, retrieved into a CR and promoted through various movements (having arranged that every file in the retrieved folder had been changed).

The test was repeated with the folder located on an external NT server.

The following table tabulates the results:

Activity	Local IFS	External NT Server
Folder Registration	4 Mins (150 MB / Min)	20 Mins. (30 MB / Min)
Retrieval to CR	7 Mins (86 MB / Min)	7 Mins. (86 MB / Min)
Check to Ready for Testing	< 1 Minute	< 1 Minute
Promotion to Module/Int	15 Mins (40 MB / Min)	30 Mins (20 MB / Min)
Promotion to Acceptance	10 Mins (60 MB / Min)	20 Mins (30 MB / Min)
Promotion to Live	20 Mins (30 MB / Min)	54 Mins (11 MB / Min)
Redevelopment	37 Mins (16 MB / Min)	100 Mins (6 MB / Min)
De-registration	3 Mins (200 MB / Min)	3 Mins (200 MB / Min)

The following points are noteworthy whilst considering the above results:

- The tests were carried out on an iSeries server model 9406-520 feature code 7450 (P05). The iSeries and NT server were connected via a wired 10 Mbps local area network;
- At the time the tests were carried out the network and machine loading was otherwise low;

- All the tests were run in batch and in a serial fashion;
- The architecture is such that retrieval times will be the same whether working in the 5250 environment or in WDSc;
- The tests were extreme insofar that SEE/Change was contrived to think that every object in the folder had changed, meaning all 602 MB were redelivered to every environment. This was deliberate to provide some indicative statistics but is not typical;
- The test bed (the iSeries server, NT machine and network) is relatively underpowered when compared to typical production environments;
- The testing did not incorporate a production system. However, because movements are conducted in much the same way on a production system, similar performances can be expected;
- The movements to the module/integration testing environment are expectedly longer than the subsequent movements to acceptance and live. This is because check-in will have occurred as part of this movement.
- Similarly the movements to live are also longer because archiving and register update will have occurred as part of this movement.

5.41.5.7 Authorisation Required for Registration

It is the registering user's authority that applies when inspecting and registering new folder. It is therefore necessary to ensure that a profile with sufficient authority is used to ensure a clean registration.

SEE/Change will report authorisation issues encountered during the registration process. However, it is important to understand that authorisation issues will only occur when the user is authorised to see a file or subdirectory but is not authorised to open it. If the user is not authorised to see the file or subdirectory in the first place, the registration process will be complete normally but the register will be incomplete.

Generally it is recommended that a special profile that is assured sufficient access be used when registering folders. For IFS folders QSECOFR or a profile having *ALLOBJ special authority is ideal. For folders on external servers, the profile being used to perform the registration must also be set up on the external server and have full access to the folder being registered.

Thenon recommend that the external server be set up with a QSECOFR administrator profile for registration and movement purposes (See section 5.41.5.9).

5.41.5.8 Authorisation to the IFS for the Movement Mechanism

As shipped, SEE/Change's MOVCROBJ program adopts QSECOFR's rights in order to ensure delivery of library objects without encountering authority issues.

For delivery to the IFS, SEE/Change uses an internal API profile handling mechanism to gain QSECOFR's rights over the IFS.

5.41.5.9 Authorisation to NT Servers for the Movement Mechanism

SEE/Change uses a profile handle to gain QSECOFR's rights over the IFS. Therefore, when delivering to an NT server, the NT server will receive an authentication request for QSECOFR rather than the user who is actually running the movement job. Consequently, a user called QSECOFR will need to be established on the NT server having the same password and sufficiently high authority to enable delivery of objects into any of the registered folders.

If you wish to use a different user profile for the purposes of delivery to NT machines, then it will be necessary to create this profile on the iSeries and change the SEE/Change MOVCROBJ program to be owned by that profile instead. However, the consequence of doing this will be that that profile will require QSECOFR equivalent rights to ensure ongoing problem free delivery of IFS and QSYS file systems objects.

5.41.5.10 Release Packaging

When release packaging, only the latest version of each stream file is packaged. The practical consequence of this is that a release must be considered holistically. That is, the entire release should be delivered and, if redevelopment is necessary, the entire release should be pulled out.

To demonstrate, consider the following example:

- A folder is retrieved to CR1 and FILEA is changed. The CR is assigned to Release 1 and put live. FILEA is now at V002.
- The same folder is retrieved to CR2 and FILEA is changed again. The CR is assigned to Release 1 and put live. FILEA is now at V003.
- Release 1 is sent to a production system and installed. FILEA will be delivered twice, once for each CR, however because the archive is single level within the release packet, it is actually V003 that is delivered for both CR1 and CR2.
- On the production system CR2 is reverted but CR1 is left at live. FILEA will still be the V003 version. If CR1 is then reverted as well, FILEA will revert to the V001 version.

5.41.5.11 Archive file changes

For performance reasons the register and archive file (O#IARC) has been changed in release 4.5003. If you used IFS Support in previous releases, your old O#IARC file will still work because the format level identifier has not changed. However, you may wish to upgrade your archive files to take advantage of the performance gains.

The file changes are as follows:

- The file has been changed to reuse deleted records i.e. REUSEDLT(*YES)
- The key has been changed as shown in the following table:

SEE/Change – Supplementary Manual 4.5003

Key prior to 4.5003 Key at 4.5003	
	IADCSY
IALEVL	IALEVL
IAMBRN	IAMBRN
IARSEQ	IARSEQ

To upgrade your archive file(s), simply replace your old archive file with a copy of the new file from your SEE/Change data library (usually OMSDTA) retaining your data to suit. le:

CRTLIB LIB(templib)

MOVOBJ OBJ(myarclib/O#IARC) OBJTYPE(*FILE) TOLIB(templib)

CRTDUPOBJ OBJ(O#IARC) FROMLIB(OMSDTA) OBJTYPE(*FILE) TOLIB(*myarclib*)

CPYF FROMFILE(*templib*/O#IARC) TOFILE(*myarclib*/O#IARC) MBROPT(*REPLACE)

DLTLIB LIB(templib)

5.41.5.12 Folder and File Name Limitations

SEE/Change supports the published NT limits of 255 characters for folder and file name lengths. However, when constructing a directory structure that is to be change managed, it is worth considering that the limits imposed by the operating system are constrained by intermediate processes.

For instance, when a file is retrieved to a CR and opened in the LPEX editor, the workbench constructs a folder path that addresses a file in the *My Documents* area of the local drive, such as that shown below:

```
C:/Documents and Settings/Marc/My
Documents/IBM/wdsc/workspace/RemoteSystemsTempFiles/194.205.2
42.98/tmp/qzrcsrvs.quser.842457.seechange/dev/cr00103302/p000
0061/public_hmtl/company.htm
```

Consequently it is possible to exceed the NT limits during normal operation even when the specific folder / file name is within the NT limits.

Generally, the rule should be to keep folder and file names sufficiently descriptive to be useful but not so long / deep as to approach the limits when prefixed in the manner exemplified above.

If SEE/Change encounters a folder path / file name that exceeds the limits during registration or movement operations, OME5255 reason code 2 will occur.

5.42 Document Library Object (DLO) Support Withdrawal (4.5003)

Document Library Object (DLO) support has been withdrawn in this release.

6 Problem Manager

6.1 Investigation request panel changes (4.3003)

When an investigation request is closed and referenced to another investigation request, the 'referred to' IR is now shown on the first and second detail panels of WRKINVRQS.

A new feature is the 'Referred to by' IR which shows any other IR's that have been closed and actually refer to the current one being displayed (this is only visible on the second detail panel).

The following example is a panel showing IR 100255 which has been closed and referred to IR 100254. it also shows that IR 100256 currently refers to 100255 (The current one).

```
SY1 Test Development System

Work with Investigation Request Details

Request number . . . . . . : 100255 Entered by: MARTIN 3/05/96 10:40:31

Menu & option . . . . . : 100255 Entered by: MARTIN 3/05/96 10:40:31

Menu & option . . . . . : / / /

Job details . . . . . . . : / / /

Dump taken ? . . . . . . : / / /

Dump taken ? . . . . . : *NO *YES/*NO

User reference . . . . . :

Notify text changes ? . . : *NO *YES/*NO

Current status / date . . : *REF - Closed/Ref: Referred to other IR 3/05/96

Referred to by IR . . . : SY1 100256

Referred to IR . . . . : SY1 100254

F1=Help F3=Exit F9=Cmd F12=Cancel F16=Bypass
```

The panel can show up to four IR's that refer to the current one. If more than four IR's refer to the current one the panel will simply take the first four IR's in descending sequence.

6.2 Replacement for Office Vision (4.4202)

6.2.1 Introduction

Currently SEE/Change has an interface to allow users to work on SEE/Change documentation using the Office Vision editor over QDLS folder documents. The documents involved are CR Developer text (Dxxxxxxx), IR user text (USxxxxx), IR support text (ISxxxxx) and Release text (RLxxxx). With the introduction of V5R1, IBM support for the Office Vision product has been withdrawn, and so the Office Vision editor is no longer available. SEE/Change's current support for QDLS folder documents has been replaced in PE 4.4202 with support for IFS files. The IFS files can be edited using a PC word processing package such as Microsoft Word.

Support for the IFS was introduced in SEE/Change with PE 4.4200 (See PE Notes for more details). SEE/Change reserves a top-level directory for each configured system using the naming convention /see#sys (where see is the configured *IFS Work Directory Prefix* and sys is the configured system name - @IFP from WRKPRMDTA). The documentation files will be placed in a subdirectory /see#sys/documents. SEE/Change will create the files and/or inform the user that they exist. The files will be created by copying template files that the user is expected to provide. A PC editor will not be invoked automatically – the user is expected to edit the files from another window, however an exit program is supported, so that, if desired, the user can write their own code to invoke their PC editor automatically via a remote command.

Note: With PE 4.4202, SEE/Change support for the Office Vision interface has been withdrawn. Although support for the IFS is functionally similar, there are some features that have been withdrawn. In particular, when packaging a Release, Office Vision documents were converted into text source members in file OMSTXT and included in the Release packet. This is not supported for IFS files.)

6.2.2 Work with OMS text Documents (WRKOMSDOC)

	Work with OMS	text Documents	(WRKOMSDOC)
Document name			
Document type		*BRW	DV, IS, US, RL *BRW *EDT *DLT *RTV
Originating system		*LOCAL	Name, *LOCAL
Target life name i Target library nam	e for *RTV	*NONE	Name, ^NONE Name, *NONE

SEE/Change – Supplementary Manual 4.5003

The Work with OMS text Documents (WRKOMSDOC) command has been amended to cater for the new IFS files. The document name has been increased to 40 characters to allow a meaningful name to be given to the files (e.g. User text for IR 000104.doc). Also a new parameter Document Type has been added to identify the four different types supported (DV=Development, US=User, IS=Support, RL=Release).

WRKOMSDOC is issued when various options are typed against IRs and CRs (e.g. 27=User Text in Work with Change Requests). If the parameter @WRD (Word processing facility) is set to "*OFC", SEE/Change will issue one of the following execution messages:

OMX0475	WRKOMSDOC DOC('Developer text for CR &13.&14.doc') TYPE('DV') ACTION(*EDT)
OMX0505	WRKOMSDOC DOC('User text for IR &13.doc') TYPE('US') ACTION(*BRW)
OMX0506	WRKOMSDOC DOC('Support text for IR &13.doc') TYPE('IS') ACTION(*BRW)
OMX0507	WRKOMSDOC DOC('Developer text for CR &13.&14.doc') TYPE('DV') ACTION(*BRW)
OMX0514	WRKOMSDOC DOC('Support text for IR &13.doc') TYPE('IS') ACTION(*EDT)
OMX0515	WRKOMSDOC DOC('User text for IR &13.doc') TYPE('US') ACTION(*EDT)
OMX0956	WRKOMSDOC DOC('User text for IR &13.doc') TYPE('US') ACTION(*BRW) SYSM(&11)
OMX0957	WRKOMSDOC DOC('Support text for IR &13.doc') TYPE('IS') ACTION(*BRW) SYSM(&11)
OMX0958	WRKOMSDOC DOC('Release text for Release &15.doc') TYPE('RL') ACTION(*EDT) SYSM(&11)

Note: Messages assume that the user will be using Microsoft Word (the files are suffixed with .doc). If a different file suffix is required (or different file names are preferred), amended versions of these messages can be created in message file OMSMSGU, however the IR number, CR number or Release number must directly precede the file suffix.

Before using WRKOMSDOC, template documents must have been created in directory /see#sys/documents (where see is the configured *IFS Work Directory Prefix* and sys is the configured system name). The templates must be named:

5	THNDVTPL.xxx	Developer text
1	THNUSTPL.xxx	User text
1	THNISTPL.xxx	Support text
5	THNRLTPL.xxx	Release text

where xxx is the word processor's file suffix (e.g. doc)

WRKOMSDOC firstly checks whether a file with the Document Name already exists in */see#sys/documents*. If found, it issues a message that the file already exists and checks whether a documentation exit program (O#DEXT) exists in the library list. If so, it calls the exit program passing the following parameters:

System	3A	
IR number	6A	(for Developer, User and Support text)
CR number	2A	(for Developer text)
Release number	5A	(for Release text)
Directory	20A	(e.g. /see#sys/documents)
Document name	40A	(e.g. User text for IR 123456.doc)
Action	4A	(*BRW or *EDT)

If the file does not already exist, it is created by copying the appropriate template (see above). A message that the file has been created is issued and the documentation exit program (O#DEXT) is processed as above.

6.2.3 Sample Documentation Exit Program (O#DEXT)

/* Sample Documentation Exit Program (O#DEXT) calling Winword using */ */ /* STRPCCMD (and Client Access) PGM PARM(&SYSM &IRNO &CRNO &RLNO &SEEDIR &DOCNAM &ACTION) /* (ignored) DCL VAR(&SYSM) TYPE(*CHAR) LEN(3) */ DCL VAR(&IRNO) TYPE(*CHAR) LEN(6) /* (ignored) */ /* (ignored) TYPE (*CHAR) LEN(2) */ DCL VAR(&CRNO) DCL TYPE(*CHAR) LEN(5) /* (ignored) */ VAR(&RLNO) VAR(&SEEDIR) TYPE(*CHAR) LEN(20) DCT DCL VAR(&DOCNAM) TYPE(*CHAR) LEN(40) VAR(&ACTION) TYPE(*CHAR) LEN(4) /* (ignored) DCL */ VAR(&SYSNAME) TYPE(*CHAR) LEN(10) DCL VAR(&PCCMD) TYPE (*CHAR) LEN (132) DCL /* Fetch the name of the AS/400 we are working on */ RTVNETA SYSNAME (&SYSNAME) /* Ensure that PC Organizer is running */ STRPCO PCTA(*NO) MONMSG MSGID(IWS4010) $/\star$ Call Winword on the PC passing parameter document name in the form * / /* "\\SYSNAME/see#sys/documents/User text for IR 123456.doc" * / /* (Note: Map a network drive on the PC to \\SYSNAME\see#sys) */ VAR(&PCCMD) + CHGVAR VALUE('c:\progra~1\micros~2\office\winword.+ exe' *BCAT '"\\' *TCAT &SYSNAME *TCAT + &SEEDIR *TCAT '/' *TCAT &DOCNAM *TCAT '"') PCCMD(&PCCMD) PAUSE(*NO) STRPCCMD

ENDPGM

7 Change Manager

7.1 Database Triggers (4.3000)

Database triggers are now handled in the same way as members and journals. i.e. function DUPDTAMBRS will preserve any triggers on promotion to the live environment.

7.2 Cross Referencing (4.3006)

SEE/Change now has a comprehensive cross referencing system built into the product. The cross reference data is collected for each configured environment allowing drill down analysis at an environment level. The data can also be used to automatically compile related objects on promotion. Therefore when changing a physical file it is no longer necessary to retrieve all related programs or even logical files into the CR. SEE/Change will automatically recompile all affected objects. This reduces concurrent development issues and ensures that a CR contains only the changes required to the application.

Cross referencing is active only on the development system. When sending releases to remote machines, relationships are resolved when the release packet is built on the development machine. Therefore the release packet contains all affected objects. Since Cross referencing is not active on a production system there is no requirement to install this enhancement on production machines in order to benefit from its functionality.

7.2.1 Switching on

The new cross referencing system can run in one of two modes. Either '*FULL' where data is collected and dependant object recompiles are executed automatically. Or if you just want to benefit from the cross reference information, mode '*DATA' will collect the cross reference data but will not execute automatic compilations.

This switch can be found in application configuration under *F18=Extended options*. Select the required mode of operation in the field *Cross referencing utilisation* valid values are *NONE, *DATA or *FULL.

Note: If you intend to use *FULL mode, you must have your application configured to *compile at development centre* *YES.

7.2.2 Initial cross reference data population

When commencing use of the cross reference facility, a once only job needs to be run to build the initial cross reference data. Command Refresh cross references (REFXREF) is supplied to do this task. The command defaults to build data for all configured applications and all libraries within them. Prompt the command to change the defaults Note the job must be run with a library list that includes your application libraries. The easiest way to acheive this is to submit the job from within a CR.

Note: Do not attempt to build cross reference data for CR libraries. Cross reference data is not held at CR level. Running *Refresh cross references* for all libraries in an application will scan the Module, Acceptance and Live program and database libraries.

This could be a long running job and we suggest you allocate a suitable time frame for the process. However, you can stage the task by running *Refresh cross references* for different libraries and or applications at different times.

From now on cross reference data will be refreshed each time an object is delivered to a new library, there is no need to ever run the full refresh again. So the cross reference data is dynamic and always correct when viewed. There is no need to schedule further cross reference updates.

7.2.3 What's new to see?

This section describes the new panels and functions that have been introduced with cross referencing.

7.2.3.1 The object browser

The object browser provides the main enquiry to the cross reference information. It is accessible via the *Work with application parts using SCDM* panel via option *26=Obj Browser*. The following panel shows the main browser panel when option *26=Obj Browser* is taken against the RPG program OMS550.

	THENON Change Management system Object Browser	View: Uses
Object OMS550 Level (P) *BAS	Type (P) *PGM Attr Environment(P) *LIV	c(P) RPG
Type options, press Ent 1=Retrieve 2=Change 12=Browse	er. 4=Delete 5=Display 7=Toggle Compile	8=Notes File Usr
Opt Object Type	Description	Usage Def Cmp
OMS933C CLP OMS983C CLP THNLVL XAO PF XAPL01 LF XASL02 LF XASL10 LF XCR PF XCRL06 LF	Create Data are LOCPRM in QTEMP RTVMBRA - Retrieve Source Member (OBJECT NOT ON OBJECT REGISTER.) Application Object Register XAP - by Dcsy/Appl XAS - by Appl/Objt/Obja/Srct/Srcq XAS - by Appl/Srcf/Srcl/Objt/Obja/S Change Request Master File XCR - by Sysm/Irno/Seqn (Shared acc	OU OU I I S I U C I More
		More
F1=Help F3=Exit F4=Pr F9=Command F10=Toggle	ompt F5=Refresh F6=Create User Defir Used/Used by F11=View	ned relation

OMS550 then populates the top portion of the panel with its attributes. Also the default environment of *LIV (Live) is selected. On the very top right of the panel the term *uses* is stated. This shows that the listed objects in the subfile are used by OMS550. So reading the panel would be "OMS550 uses"

The example above shows a couple of programs that are called by OMS550 and a set of physical and logical files. The usage column indicates whether this file is used by OMS550 for Input, Output etc. So file XAO is updated and written to by OMS550.

You can toggle the *uses* to *used by* by pressing F10, this will then show the reverse of the coin in that it will show the objects that use OMS550, as the following example shows.

	THENON Change Management system Object Browser	View: Used by
Object OMS550 Level (P) *BAS	Type (P) *PGM Att Environment(P) *LIV	r(P) RPG
Type options, press En 1=Retrieve 2=Change 12=Browse Opt Object Type	er. 4=Delete 5=Display 7=Toggle Compile Description	8=Notes File Usr Usage Def Cmp
MOVCROBJ CLP	MOVCROBJ - CPP	
		Bottom
F1=Help F3=Exit F4=P: F9=Command F10=Toggle	compt F5=Refresh F6=Create User Defi Used/Used by F11=View	ned relation

The panel is now telling you that OMS550 is *Used by* CL program MOVCROBJ. Pressing F12 from here will return you back one step to the OMS550 *Uses* panel.
There are two ways to turn your attention to another object, you can either type over OMS550 at the top of the panel with the new object, type and attribute etc or you can simply take an option *12=Browse* against one of the objects listed. Function key F12 will always bring you back 1 step so no matter how many times you change the key at the top or take option *12=Browse*. F12 will return you one step at a time back to your initial screen. This facilitates tree searching as you can reverse and take another branch.

Basic Options:

- Option 1=Retrieve will bring the selected object into the CR where you launched the browser from.
- Option 5=Display will present a small menu of options that are available for the selected object. For instance the ability to view source, look at the objects description, view data members or even give a module break down for ILE objects. the available options shown on the menu are dependant on the object type selected.
- Option 7=Toggle compile may sometimes be necessary to prevent an automatic compile from taking place. Using this toggle will prevent the selected object being recompiled. an *NO appears under the Cmp column for objects that are prohibited from recompiling. This is a toggle option so selecting option 7 again will remove the prohibit compile symbol.
- *Option 12=Browse* allows the selected object to be made the focus of attention.

7.2.3.2 User Defined relationships

You may wish to set up your own relationships. This may be done for one of two reasons.

- You wish an object to recompile that cross referencing would not know about because they don't actually reference each other within the same environment.
- You require a notational link. le a text based link between two objects that does not cause a recompile of the related object. Using this method you could warn a programmer of related programs when s/he is retrieving items into their CR by way of a note.

To create a new relation press function key *F6=Create User Defined Relation*. The following panel appears:

	THENON Change Management system. Add User Define Reference
	Dev Centre Sys : THN Thenon Development base Appl Code : OMS Thenon
Object NEWP Object Type . *PGM Object Attr RPG Source Type . *BAS Object Lib *LIB	<pre>GM == USES ==> Object Name : OMS550 a == USES ==> Object Type (P): *PGM a== USES ==> Object Attribute (P): RPG a== USES ==> Source Type (P): *BAS L == USES ==> Library Name : *LIBL</pre>
User Defined Record. Environment	<pre>(P): NOTES Notational Link (P): *ALL All Environments</pre>
F3=Exit F4=Prompt	F8=Notes F9=Command F10=Toggle Use F12=Previous

On entry to this panel defaults have been set up to say that the new object you are about to enter *Uses* OMS550. This is because we pressed F6 on the OMS550 *Used By* panel.

Entries have been made to say that NEWPGM uses OMS550 the link is in notes form and applies to *ALL environments. Remember all cross reference data is held at environment level allowing certain compiles to action or not action. Therefore you could for example set your system to recompile on promote to live but not on some promotions during testing.

Pressing enter on this panel will present a window where text notes can be entered. The notes as well as being accesible from the object browser also may pop up during a retrieve into a CR. Ie if a programmer retrieves an object, say NEWPGM, they will be presented with a message that says, "NEWPGM is linked to object OMS550, would you like to view the notes?" answering yes will cause the notes window to pop up in browse mode only.

The remaining options on the main browser panel relate to the user defined relationship functionality.

- Option 2=Change to change the details of a user defined relationship
- Option 4=Delete to delete
- Option 8=Notes to view the user defined notes

Object Browser Summary

The object browser is the main enquiry for cross reference data. It shows information for all registered objects within the application at an environment level.

7.2.3.3 Dependancy Checking

You may at some time wish to check on all the dependant objects for a given CR, ie check what is going to be automatically recompiled when a promote request is actioned. This is available through a new option *46=Dependants* on the *Work With Change Requests* panel as follows:

	TH	ENON Change Dependanc	Manageme y File En	nt system quiry	
Application Change Reque	AP1 st 100	Demo app 365 / 09	lication	1	
File Name in CR	Dependant Object	Object Library	Туре	1st Occurrence	Comments
MK8711	MK87PGM MK87PGM	AP1SY1PA AP1SY1PL	* PGM * PGM	*	Not source based Not source based
MK87L2	MK87P2 MK87P2	AP1SY1PL AP1SY1PM	* PGM * PGM	*	Live source not found Live source not found
MK87PF	MK87L1 MK87L2	AP1SY1DA0 AP1SY1DA0	*FILE *FILE	*	Live source found Live source found
	MK87L1 MK87L2	AP1SY1DA1 AP1SY1DA1	*FILE *FILE	*	Live source found Live source found
	MK87L1 MK87L2	APISYIDL0 APISY1DL0	*FILE	*	Live source found Live source found More
Open of memb F3=Exit F9	er O#WRKDV =Command F	was changed 10=1st Occu	to SEQON rance F1	LY(*NO). 1=View F12=Ca	ancel

The first column *File Name in CR* shows the object that resides in the CR. The following columns show the dependant object and its library. The first occurrence asterisk merely denotes that this is the first occurrence of the given object in the given library. ie usually the same object is related to many objects and so seemingly appears many times as a requested recompile. (It is worthwhile to note at this point that the movement mechanism will compile required objects once only). The final column gives a brief note, the program will actually search for a source member, with ILE objects this will not be found.

7.2.3.4 Automatic recompilations

During a physical promotion, required compilations will be performed. Should a compile fail, the movement as a whole will fail. Rerunning the movement will as usual recover from the point of failure. A new spool file O#1610 is generated during a movement. Like OMS4400 which details movements O#1610 will detail recompiles giving error messages for failures.

Another way to view recompilation status after a movement is via another new option on *Work with Change Requests 47=Recompiles*. This panel follows:

	SY1	Test Devel Recompilati	opment System on Enquiry	stem Y			
Type options, 5=Display	press Enter		Change Red	quest	. 100365 /	09	
Opt Object	Attribute	Target Library	Source Member	Move Type	Oper Stat	Compiled Date	
MK87P2 MK87L1 MK87L1 MK87L2 MK87L2 MK87P2 MK87P1 MK87P1 MK87L1 MK87L1 F1=HELP F3=Ex	RPG LF LF EF RPGLE_SRC RPG RPG LF LF LF	APISYIPM APISYIMDL APISYIMD2 APISYIMD2 APISYIMD2 APISYIPM APISYIPM APISYIPM APISYIPM APISYIMD2 sh F9=Comm	MK87F2 MK87L1 MK87L2 MK87L2 MK87L2 MK87P1 MK87F1 MK87F1 MK87L1 MK87L1 MK87L1	*MDL *MDL *MDL *MDL *MDL *MDL *MDL *RST *RST Cancel	*CPL *E66 *CPL *OK *CPL *OK *CPL *OK *CPL *OK *CPL *E61 *CPL *E66 *CPL *CHK *CPL *CHK	00/00/00 13/03/97 13/03/97 13/03/97 21/02/97 00/00/00 00/00/00 12/03/97 12/03/97 More	

As in the example program MK87P1 failed to compile giving a failure code of *E61. Taking option *5=Display* will give more details as follows:

	SY1 Test Development System Recompilation Enquiry	
:	Object Compilation Details	:
:		:
:	Object Compiled : MKRCP1 Date : 00/00/00	:
:	Type: *PGM Compile Time:	:
:	Attribute : RPG Compile Job No :	: d
:	Compile Job Name :	:
:	Target Library : AP1SY1PM User/Group Profile. :	:
	Source Library : VP1SY1POOL Source Type / Oual. : *BAS	: 0
	Source File : ORPGS01 Site Code :	: 7
	Source Member	• 7
	Module Source Name : Operation : *CPL	• 7
:	Completion Status : *E61	• 7
:	Error changing LTBL for compile	≏ • 7
:	Recompiled Due To Related Object .	• 0
:	Object Moved · MK87DE Turpe · *EILE	. 0
:	Attributo · PE Transaction No · 00001476	2 . 7
:	E2-Evit E2-View Bolated Object Mewements E0-Command E12-Cana	∠ • / ~1 • 7
:	ro-txit ro-view keialed object Movements ro-command riz-cance	er • /
÷		• • •
:.	•••••••••••••••••••••••••••••••••••••••	:

This gives details of the compilation request, it also shows (In the last two lines) the object that caused the recompile (Obviously if many objects referred to this object only the first is shown). The compilation status *E61 is explained as a library list problem, most like due to a missing job description in the target library. File MK87PF was the object that caused MK87P1 to be recompiled. Pressing *F8=View Related Object Movemements* at this point will take you to the movement history panel for the file MK87PF.

7.3 Library creation exit program (4.3006)

A new hook has been put into SEE/Change whenever a library is created. Immediately after creation of a library SEE/Change will look for a program called O#AEXT. If found it will be called passing parameters (LIB, TYPE, TEXT) the same as the CRTLIB command itself.

7.4 Year 2000 Compliance (4.4000)

This release of SEE/Change has been fully tested to operate normally before, during and after the transition into the year 2000.

7.5 New exception report for cross referencing (4.4001)

The cross-referencing process produces a new report called O#159P. If exceptions are encountered during the cross-referencing a brief description of the exception and possible corrective action is detailed in the report.

7.6 New user exit program (4.4001)

A new exit program has been plugged into the receive release (RCVRLS) mechanism just before the validation and movement from the release packet takes place. The system will look for a program called O#MEXT and if found will execute it passing the following parameters:

System	3A
Release No.	5A
Release Library	10A
IR / CR No.	8A
Error Flag	4A

You must populate the result flag (i.e. '*OK'). If the result flag comes back as *BLANK then the movement will terminate.

7.7 Changes to ILE processing (4.4002)

Changes have been made to the way SEE/Change handles ILE objects. A new function has been created called *Assemble ILE Program* (ASSILEPGM). This new program rebinds all component modules into a given program.

The new function is run whenever a program is moved to a new environment or whenever a module is moved that affects programs in other CR's. In the case of concurrent development of ILE programs where multiple programmers have different component modules from the same program in their own CR, and the CR's are at varying stages through the lifecycle. The new function will ensure that a moved program contains the correct modules for the environment that it has been moved to.

If another programmer reverts a CR containing a module then all the programs in other CR's that make use of the module in that and lower environments will be updated with the correct version (i.e. the old version).

7.7.1 Changes to the OMS440 report

The movement log report OMS440 has been enhanced to show the activity of the *Assemble ILE program* process. Therefore it is now possible to see the exact bindings that have taken place due to an object movement.

If cross-referencing is used to execute recompilations then OMS440 now also shows the recompilations for a given object. This has replaced the O#161 report that previously listed recompilations separately.

Both the above details are given directly after the line in OMS440 that triggered them.

7.7.2 Changes to application configuration

The Assemble ILE program process attempts to rebind all constituent modules. The program will fail and therefore halt the movement if an expected module is not found. If you do not have a complete set of modules on your remote machines you need to bypass this error. Therefore a new flag has been included in the application configuration to allow you to ignore missing module errors.

You should never ignore missing module errors on the development centre as it signifies an incomplete set of application objects. See the help text on the field for more information.

7.7.3 Changes to application set up

An application job description is required in all database libraries. A job description in a program libraries was only necessary if your promotes recompiled all objects.

Due the rebinding process now executed on ILE programs it is a prerequisite to have the job description in the program libraries also, whether compiling or not.

7.8 Bound programs meeting in an environment (4.4002)

When and ordinary OPM program (i.e. RPG) program is promoted to module or acceptance and there is another version of the same program already residing, SEE/Change offers to replace the existing object.

Due to the complexity of ILE programs this facility was not implemented for any kind of ILE program. This rule has been relaxed for people creating only bound programs (ILE programs created from a single source) so that you can now replace the existing object as before like the OPM processing. However SEE/Change will still prevent multi module programs being replaced in this way.

7.9 Changes to controlled deletions (4.4002)

Previously when a database object was marked for delete it would not actually be removed from the module and acceptance environments until the change request was promoted to live.

This has been changed. The database object will now be deleted in each environment as the change request is promoted through to live.

7.10 Check CR Validation to *TST or *RLS (4.4003)

Previously when moving a CR to status *TST, determining what the actual errors were when the job was submitted to batch involved examining the job log. Now a separate spool file is produced which itemises the errors raised.

7.11 SQL Support (4.4300)

7.11.1 Promoting / Reverting SQL-based objects

When promoting a CR containing an SQL-based object, the movement mechanism processes the associated SQL source in a similar way to the way that it is handled in the CR. A temporary copy of the source member is made and amended so that the primary object being referenced is qualified with the target library for the environment being promoted to. The RUNSQLSTM command is then issued against the temporary source member.

The target libraries for tables, indexes, views and triggers are the configured application data libraries.

The target libraries for procedures and functions are the configured application object libraries.

The Movement Operation for SQL-based object types will be reported as '*SQL' (e.g. on the Movement Logs Listing: OMS440).

7.11.1.1 SQLTABLE, SQLINDEX, SQLVIEW

The movement of a table, index or view is almost identical to that of a physical or logical file. If an existing file is found in the target library, a work library is created, the existing file is moved into it, the SQL source is run to create the object and, if appropriate, data is copied over using CPYF.

If this is a live movement and archiving is active, the work library is archived, otherwise the file is deleted and the work library removed.

If the CR is reverted, the Live version of the file is replaced by the archived version and the Acceptance and Module/Integration versions are reinstated by either duplicating the replaced Live version or by issuing a RUNSQLSTM against live source.

7.11.1.2 ALTER TABLE

For a movement of a table that is to be modified using ALTER TABLE, a work library is created and populated with a copy of the table in the target library but, importantly, *without* data. The modified ALTER TABLE source is then run against the table in the target library. Again, if this is a live movement and archiving is active, the work library is archived, otherwise the table is dropped and the work library removed – i.e. the archive for an ALTER TABLE movement is an empty copy of the unaltered table.

If the CR is reverted, the current make-up of the Live version of the table is compared to the archived version and a new ALTER TABLE statement is calculated and run to convert the table back to its previous state. Similarly, the Acceptance and Module/Integration versions of the table are compared to the Live version and a new ALTER TABLE statement calculated and run.

Note: For ALTER TABLE, the system will automatically handle any dependent views/indexes. If ALTER TABLE is not used for the amendment of a table, the user is expected to handle dependencies explicitly or make use of Cross Referencing.

7.11.1.3 SQLPROC, SQLFUNC, SQLTRIGGER

Procedures, functions and triggers will have an associated program or service program and come in two basic forms; external or internal.

External procedures, functions or triggers are compiled from high-level language source which is retrieved separately to the CR, that is, there is the SQL element and a secondary, HLL element.

Internal procedures, functions or triggers, are written in procedural SQL and created by the system when the *create* procedure, function or trigger source is run.

In either case, the movement mechanism does not explicitly handle the associated program / service program when moving procedures, functions and triggers. It simply runs the SQL source member at the appropriate environments to create the *entity* and / or issues DROP statements in order to remove the entity.

Checks for entity existence are achieved through look-ups on the system catalogues in library QSYS2.

The movement mechanism treats procedures, functions and triggers as follows:

If the entity already exists in the target library, a work library is created and loaded with a source file. The *Generate DDL* API is issued against the existing entity and the generated source placed into this source file. The existing object is then dropped and the SQL source from the CR library is run to recreate the entity. If this is a movement to Live and archiving is active, the work library is archived, otherwise the work library is removed. I.e. archives for these entities are actually stored as SQL source members.

When reverting from Live, the generated source from the archive library is run. For Acceptance and Module / Integration reversion, the live source is run.

Procedures and functions are delivered to the application object libraries; that is, procedure and function entities are treated as *program objects*. In keeping with the SEE/Change methodology for objects, when a CR containing a procedure or function entity is promoted from one environment to another, the entity is dropped from the previous environment.

Triggers are delivered to the application data libraries; that is, trigger entities are treated as *database objects*. In keeping with the SEE/Change methodology for database objects, when a CR containing a trigger entity is promoted from one environment to another, the entity is *not* dropped from the previous environment and *is* created in all intermediate environments.

7.11.1.4 SQLDML

SQLDML movements will occur after all other movements for a CR.

With SQL DML, there is no associated *object* or *entity*. During promotions and reversions the source is simply run for each data environment. The source may be organised into two sections – a section to be run when promoting forward and a section to be run when reverting. Before running the source, SEE/Change will make a temporary copy of the source and comment out the section that does not apply.

When SQLDML is reverted from live, the source pool movement is processed last. This is a subtle irregularity in the reversion sequence necessary because the "object" movements need to run the *reverse* section from the current live source member first.

At production sites, if source is not distributed but archiving is active, SEE/Change will copy the source to an archive library to enable a subsequent reversion to be processed.

7.11.1.5 Object Authority

The Apply Object Authority routine (APYOBJAUT) will be run after each SQL-based object movement in the same way as it would for any other movement. If the source for the SQL-based object contains SQL GRANT or REVOKE privileges, these may be overridden when the Apply Object Authority routine is processed. If you wish to maintain GRANT and REVOKE privileges, place these in a SQLDML source member in the CR. These will then be applied after all the object movements have been completed.

7.11.2 Further SQL related topics

7.11.2.1 SQL Long names

SQL offers support for names of up to 128 characters. However, the SQL object will always be assigned a ten-character system name (the system may generate the system name for you if you specify more than ten characters). If you use long SQL names, SEE/Change requires that you craft your SQL so that the object created will have the same system name as the source member name. For example, member MYINDEX in SQLINDSRC might contain the SQL:

create index INDEX_WITH_A_LONG_NAME on MYTABLE (MYCOLUMN);

rename index INDEX_WITH_A_LONG_NAME to system name MYINDEX;

Having executed the above SQL statements the new table will be accessible through SQL via the long name or the short name and through non SQL interfaces via the *short name* of *MYINDEX*.

7.11.2.2 Concurrent development

Concurrent development of SQL-based objects is not allowed.

7.11.2.3 Importing SQL based objects

The IMPORTLIB function has been enhanced to support SQL object types. To import SQL establish source along with objects where appropriate in your import library and issue the IMPORTLIB command in the same manner as with previous versions. See PE Note 4.4202 for full details on IMPORTLIB and IMPORT.

For SQLTABLE, SQLVIEW, SQLINDEX, the import process will import both the source and the generated object.

For SQLPROC, SQLFUNC, SQLTRIGGER and SQLDML, the import process will import the source only.

The source for the SQL-based objects must reside in members in source files in the external library as per your application configuration. The table in section 5.23.2.2 shows the *default* source file names and member types that SEE/Change uses.

7.11.2.4 SCNDBREL compatibility

For tables, views and indexes Scan Database Relations (SCNDBREL) is supported so that dependent parts can be identified and action taken to suit.

7.11.2.5 Cross Referencing

For tables, views and indexes, cross referencing will, if enabled, recompile dependent parts upon promotion in the same way that it does for physical and logical files.

7.12 Database Trigger Support (4.4301)

7.12.1 Introduction

A trigger is an automatic call to a program that is invoked when a specified event occurs on a database physical file. The event can be a read, insert, update or delete.

A trigger may be created either by SQL statement *CREATE TRIGGER* or by CL command *ADDPFTRG*, Add Physical File Trigger.

In the past two issues have affected the use of these triggers in SEE/Change:

- During promotion, when an existing physical file with a trigger is replaced by a new version of the file, the trigger is not re-applied to the new version of the file;
- If a file with a trigger exists at a given environment, and we promote a trigger program to that environment, then we would expect the promoted version of the program to be called when the trigger fires. However, the system will call the version of trigger program in the library declared when the trigger was added to the physical file (normally the live object library). This is because the location of the trigger program is fully qualified at creation time, and means that an amended trigger program will not be called until it is promoted to the declared trigger program library.

Both of these issues have been addressed in PE 4.4301.

7.12.2 Preserving existing trigger definitions

When a file is moved through the CM lifecycle, SEE/Change 4.4301 will preserve file trigger definitions. The processing is as follows:

- Record all trigger information for the existing file (V1);
- Disable triggers on file V1 to prevent them being fired during data copy operations;
- Move file V1 to temporary library (this becomes the archive library if movement is to live and archiving is enabled);
- Install new file (V2) and copy the data from V1 using DUPDTAMBRS;

• Add trigger definitions to file V2 exactly as per file V1.

The above processing will preserve the file trigger information precisely. Most notable is that the trigger program library will be preserved exactly as it was on the file being replaced. In other words *LIBL is <u>not</u> used to resolve to the trigger program location on a file movement.

An error in this processing will fail the movement and diagnostic messages will be listed to the movement log. The movement will get an error status.

In reversion the processing is the same:

- Record all trigger information for the existing file (V2);
- Disable triggers on file V2;
- Move file V2 to temporary library;
- Restore file V1 back to environment library (this is taken from the archive library if reversion is from Live, and from the live library if the reversion is from Acceptance or Module. Reversion from Live automatically encompasses other environments);
- Copy the data from V2 to V1 using DUPDTAMBRS;
- Add trigger definitions to file V1 exactly as per file V2.

7.13 Cross Referencing (4.4301)

7.13.1 Introduction

SEE/Change's cross-referencing facility was introduced with PE 4.3006. Please refer to PE Notes 4.3006 for a full description of crossreferencing.

With PE 4.4301, various enhancements have taken place.

- The object browser has been made accessible from the *Display CR Parts* display at development centres;
- For referencing ILE objects, the object browser now displays both modules and programs;
- The object browser has been reworked to make it easier to identify the level (*BAS/*GRP/*SIT) and environment of cross-referenced objects;
- Group (*GRP) and site (*SIT) level objects are now fully supported;
- The release packaging process has been enhanced to support naming conflicts at differing software level;
- Additional controls have been built into the movement mechanism to ensure errors occurring in cross referenced recompiles are reported.

7.13.2 Object Browser

The object browser is now available from the *Display CR Parts* display:

		Chan	nge Management Display CR Par	System ts		7
CD . CV1 000	0007 / 00	0	den Debuild		1	Snvr: *LIV
CR : SIL UUU	JUU7 / UZ	01	aer Rebuild			
Appi : API Den	10 APPI 4/11/02		niaci Rei:	MD D++	• * \$711T	D1c. 00521
E Diselar	4/11/03 11 mm	AS	Signed to: QPC	MR PLY	: ^VH1	RIS: 00521
25-DISPIAY	26-Obi Pr	Let	20-Movements	ZI-HISLOI	у 22	2-Overrides
Opt Object P	Pof Id (P)	Toyt				
opt object h	Ner Id (F)	TENC				
ORDHDRL1 I	मन	Order	Header			
ORDF01 M	MNUDDS	Order	Menu			
26 ORDHDRP P	PF	Order	Header			
ORDF01 M	MENU	Order	Menu			
ORDHP00 R	RPGLE MOD	Order	Test		Marked	1 Delete
ORDHP01 R	RPGLE MOD	Order	Header Display	7		
ORDHP12 R	RPGLE MOD	Order	Clients			
ORDHP13 R	RPGLE MOD	Order	Accounts			
ORDHP14 R	RPGLE MOD	Order	History			
ORDHP15 R	RPGLE MOD	Order	Summary			
ORDHP16 R	RPGLE MOD	Order	Related			
						More
F3=Exit F5=Ref	fresh F9=C	Cmd F1	1=Change view	F12=Cancel	F19=IFS	objects

The object browser has been enhanced to show module parts as well as program parts. Also a new column to show the level at which the referencing part is at has been included:

	Change Management System Object Browser	View: Used by
Object <u>ORDHDRP</u> Level (P) <u>*GRP</u> <u>STH</u>	Type (P) <u>*FILE</u> Environment(P) <u>*LIV</u>	Attr(P) <u>PF</u>
Type options, press Enter 1=Retrieve 2=Change 4= 12=Browse	:. Delete 5=Display 7=Toggle Re	build 8=Notes Re-
Opt Object Type	Level Description	Env bld
ORDHP01 RPGLE_MOD ORDHP01 RPGLE ORDHP02 RPGLE ORDHP03 RPG ORDSRVPGM *SRVPGM ORDHP01 RPGLE_MOD ORDHP01 RPGLE ORDHP02 RPGLE ORDHP03 RPG	*GRP STH Main order processin *GRP STH Main order processin *GRP STH Order notes (for STH *GRP STH Generate new order (*GRP STH Order processing srv *SIT GAT Main order processin *SIT GAT Main order processin *SIT GAT Order notes (for GAT *SIT GAT Generate new order (g (for STH) *LIV g (for STH) *LIV) *LIV for STH) *LIV pgm (for STH) *LIV g (for GAT) *LIV g (for GAT) *LIV) *LIV for GAT) *LIV More
F1=Help F3=Exit F4=Prom F9=Command F10=Toggle Us	npt F5=Refresh F6=Create User sed/Used by F11=View	Defined relation

The secondary view has also been modified slightly to accommodate these changes:

SEE/Change – Supplementary Manual 4.5003

		Change Manaq Object H	gement System Browser		View:	Used b	У
Object. <u>ORDH</u> Level (P) <u>*GRP</u>	DRP <u>STH</u>	Type (H Environment(H	P) <u>*FILE</u> P) <u>*LIV</u>	Attr	(P) <u>PF</u>		
Type options, 1=Retrieve 2 12=Browse	press Enter. =Change 4=D	elete 5=Disp	play 7=Toggle	Rebuild	8=Note	s Usr	Re-
Opt Object	Туре	Attribute	Library	Usage		Def	bld
<pre>ORDHP01 ORDHP01 ORDHP02 ORDHP03 ORDSRVPGM ORDHP01 ORDHP01 ORDHP01 ORDHP02 ORDHP03</pre>	*MODULE *PGM *PGM *SRVPGM *MODULE *PGM *PGM *PGM	RPGLE_MOD RPGLE RPGLE *SRVPGM RPGLE_MOD RPGLE RPGLE RPG	AP1STHOL AP1STHOL AP1STHOL AP1STHOL AP1STHOL AP1GATOL AP1GATOL AP1GATOL AP1GATOL	IO IO IO IO IO IO			
	1011	1010	111 1011101	10		Mo	re
F1=Help F3=Ex F9=Command F1	it F4=Promp 0=Toggle Use	t F5=Refresh d/Used by F1	n F6=Create U 11=View	ser Defin	ed rela	tion	

7.13.3 Release Packaging

When cross-referencing is set to *FULL, cross-referenced objects with the same name, but at different levels, can now be packaged into a single release packet. For example, a PF is referenced by both a *BAS and *SIT level version of a program. If this PF is retrieved to a CR and that CR is allocated to a release, then when the release packet is built, both versions of the program will be included and both will be delivered to production.

7.13.4 Additional movement controls

Previously, failing recompilations of cross-referenced parts would not cause a CR movement to fail. Controls in this area have been improved now so that a given CR part movement (for example, the physical file) is only considered to have completed successfully when all it's related cross reference parts have also successfully completed recompilation. If this is not the case the movement will be assigned an error status (or reverted if configured to do so).

As with all CR's having an error status in SEE/Change, option 11 allows the movement to be retried following remedial action, or reverted if so desired.

7.13.5 Restrictions

When cross-referencing is set to *FULL, SEE/Change will re-compile referencing objects during promotions. In general, if a referencing object is at a different level to the referenced object (e.g. a *SIT level program references a *BAS level physical file), it will be compiled correctly.

SEE/Change – Supplementary Manual 4.5003

However, there is one exception. Cross-referencing will not support the recompilation of a logical file that is scoped over a physical file at a different level (e.g. a *SIT level logical scoped over a *BAS level physical). If you have such arrangements at your installation you must handle them explicitly – i.e. retrieve them to the CR along with the related physical.

7.14 Movement reporting & diagnosis improvements (4.4301)

7.14.1 Introduction

On movement completion, SEE/Change previously provided two reports:

- movement logs listing (OMS4400)
- movement error log report

With PE 4.4301, the movement error log report has been integrated into the movement logs listing. Any messages that apply to the movement of a particular object will now be printed directly under the movement log for that object. This will make it much easier in future to identify the reasons for a movement failure.

A number of other associated changes have been made in this area:

- The LSTMVTLOG (*List Movement Logs*) command will also print the new style of report, and has new parameters to control this;
- The Change Manage has new fast track options to access movement reports.

7.14.2 LSTMVTLOG (List Movement Logs)

New parameters have been added to the command LSTMVTLOG.

List Moveme	ent Logs (LST	IMVTLOG)	
Type choices, press Enter.			
Job Name Job Number	* <u>NO</u> * <u>YES</u>	Name Name Character value Name 000001-999999 01-99 Date *YES, *NO *YES, *NO	
F3=Exit F4=Prompt F5=Refresh F24=More keys	F12=Cancel	Bottom F13=How to use this display	

If the parameter *List only movements in error*? (ERRONLY) is set to *YES, the generated report will only list movements with error completion codes (together with any associated messages). The default for this parameter is *NO, and as such the report will behave as it has done in previous releases (other than to include any associated messages).

The parameter *Include 2nd level msg text?* (SCDLVL) determines whether second-level message text is printed on the generated report. The default is *YES, meaning that second-level message text is included. If this is set to *NO then only first-level message text is printed on the report.

7.14.3 New Change Manage fast track options

To aid navigation to the upgraded reports, two new options have been added to the *Work with Change Requests* screen.

Change Management System Work with Change Requests	Filter:	*NONE
49=CR Attachmnt 65=Chk CR Log 66=Mvt Reports 67=List	Mvt Err	
Opt IR No/CR Text	Applicatn	Status
<pre>000001 Test 4.4301 environment build 01 Test 4.4301 environment build 2 02 Test 4.4301 environment build 2a 03 Test 4.4301 environment build 3 04 Test 4.4301 environment build 4 05 Test 4.4301 environment build 4a 06 Test 4.4301 environment build 4b</pre>	V4.43 test V4.43 test V4.43 test V4.43 test V4.43 test V4.43 test V4.43 test V4.43 test	Opened CRs Live/Prod Live/Prod Live/Prod Live/Prod Err/Live
		Bottom
F1=Help F3=Exit F4=Prompt F5=Refresh F6=Create F9=C F12=Cancel F14=Curr flt F21=Filter F22=Status F23=Mc	Cmd F11=Chan ore options	ge view F24=Msgs

Option 66=Mvt Reports will display the job spool files for the last CR movement job (if the spool files are still on the system).

Option 67=List Mvt Err will issue the LSTMVTLOG command for the CR's last movement with the parameter *List only movements in error*? set to *YES. The generated report will list only movements in error along with their completion code.

Thus executing option 67 against a CR that has gain an error status will yield an instant diagnosis of the problem.

7.15 Improved control of Reversion movements (4.4301)

Previously, errors occurring during the reversion of CR's did not cause the reversion movement to fail. This could result in a CR appearing to have been reverted when in fact one or more parts might remain installed in the environment libraries. PE 4.4301 addresses this by incorporating reversion movements more fully into the movement mechanism. Now, movements arising as a consequence of reversion can obtain an error code in the same was as forward movements can. Such errors will cause the CR to rest in a new *Err/Redvlp* status.

Change Management System Filter: Work with Change Requests	*NONE
2=Change3=Copy/Crt CR4=Delete5=Display8=Dis9=Rls distrib10=Network Sts11=Promote12=Wrk CR dev13=Wrk	play obj : CASE
Opt IR No/CR Text Applicatn	Status
000005Add last maintained field to item masterV4.4300 te01 Add last maintained field to item masterV4.4300 te	Opened CRs Err/Redvlp
	Bottom.
F1=Help F3=Exit F4=Prompt F5=Refresh F6=Create F9=Cmd F11=Chang F12=Cancel F14=Curr flt F21=Filter F22=Status F23=More options F	je view 724=Msgs

As with all forward movements, the reversion movement now also generates an OMS440O movements log report, in the new format that has been introduced with this release.

As with CR's having forward movement error status conditions, the reversion movement can be reattempted using option 11.

CR's at the *Err/Redvlp* status are not accessible for development.

7.16 Allow reset of Ready for Release status (4.4301)

7.16.1 Introduction

Prior to PE 4.4301, a CR that had been moved to status *Ready for Release (*RDY)* could only be either promoted to *Live (*LIV)* or reverted for *Redevelopment (*RDV)*.

The movement to *Ready for Release* does not cause any objects to be moved, it simply flags the CR as being available to be included in a Release Packet, subject to integrity and authorisation check. As such removal of the flag will revert the CR to the environment that it was in prior to being set to Ready for Release.

With PE 4.4301, a new parameter – *Allow reset of Ready for Release* (@*ARR*) – has been included; this parameter allows the unflagging of a CR that has been moved to Ready for Release without having to revert the CR for Redevelopment.

7.16.2 Configuration

The new parameter (@ARR) can be accessed via the *Work with Parameter Data* (WRKPRMDTA) screen.

		Change Management Syst	em						
2	-Chang	work with Parameter Da	та						
23	-change	е о-ызртау		Dam					
0	- 0-1-	Description		Para	nnete	er a	TDT	Juces	3
Op	t Code	Description	VAR	LEN	DEC	DSP	EDT	νгр	AUT
	*TBL	Example of user pattern table	10	40	A				2
_	@AAR	Allow Acceptance retest?		4	А			V	2
_	@AIE	Allow imported source to be edited		4	A			V	2
_	@AMM	Allow Missing Modules		4	A			V	2
_	@ARP	Archive/Work Library Prefix		1	A			R	2
	@ARR	Allow reset of Ready for Release		4	A			V	2
_	@AUD	Authorize Distribution to Systems?		4	A			V	2
_	@AUT	Authorisation Code - SEE/Change		20	A				2
	@AU1	Authorisation Code - SEE/One		15	A				2
_	@ AU2	Authorisation Code - SEE/Job		15	A				2
_	@ AU3	Authorisation Code - Shuttle		15	A				2
_	@CAS	Case Tools Import Prompt Flag Y/N	3	1	A				2
_	@CAU	*PUBLIC Authority to CR		10	A				2
_	@CDR	Check Direct to Ready		4	A			Y	2
_	@CLT	CR Library Type *PROD/*TEST		5	A				2
								Mo	re
F1	=Help	F3=Exit F6=Create F9=Cmd F11=Delete	F12=0	Cance	el E	521=1	Print	t li:	st

Option 2 allows the user to maintain its' value.

Change Management System Work with Parameter Data	:
Attr: *LEN: 0 / 4 A *VLD: Value List: *DSP: *STD *VL1: *YES *EDT: *STD *VL2: *NO	
Enter/update parameter value: @ARR : Allow reset of Ready for Rel	ease .
Parameter Value *YES or *NO	
	· ·
	· · ·
	Bottom .
F1=Help F3=Exit F9=Cmd F12=Cancel	:

If this parameter is set to *NO or left blank, then a CR that is at status Ready for Release can only be either promoted to Live or reverted for Redevelopment.

If this parameter is set to *YES, then a CR that is at status Ready for Release can be promoted to Live, reverted for Redevelopment or reset to the environment that it was at immediately prior to it being moved to Ready for Release (such as Module/Integration or Acceptance).

If 11=Promote is chosen against a CR at Ready for Release in *Work with Change Requests*, then the pop-up window – *Select Movement Type* – will include the option to move the CR back to its previous environment.

SEE/Change – Supplementary Manual 4.5003

Change Management System Filter: *NONE Work with Change Requests 2=Change 3=Copy/Crt CR 4=Delete 5=Display 8=Display obj 9 Select Movement Type Op : CR Nbr : AP1 000002 / 05 Application: Demo Appl : Status : *RDY - Rdy/Rlease: Ready for Release : CRs : Select one of the highlighted movement types, press Enter. : od Testing ...first, revert CR for development Integratn ...first, revert CR for development : od : od : : _ Acceptance Acceptance/QA library : od : Rdy/Rlease ...first, revert CR for development : Live/Prod Live/Prod library 11 : : mnt : mnt. : _ Redevelop Revert to Development : ed : od : Enter any character to select F1=Help F12=Cancel : od :..... Bottom F1=Help F3=Exit F4=Prompt F5=Refresh F6=Create F9=Cmd F11=Change view F12=Cancel F14=Curr flt F21=Filter F22=Status F23=More options F24=Msgs

7.17 Stream File Support (4.5003)

7.17.1 The Movement Process

SEE/Change performs stream file movements after completing any library object movements for the CR. Directories and files are created, replaced or deleted at the file system location defined for the target environment when the folder was registered. Additionally, for promotions into the *LIV environment, the register is updated and files that have been changed or deleted are archived in a manner analogous to that for library objects. These archive versions are used in any subsequent reversion.

For folders with environments defined on NT servers, SEE/Change updates the appropriate network share. This implies that the network share must be visible in /QNTC at promotion time, otherwise the promotion will fail.

8 Development Manager

8.1 Changes to ILE processing (4.3000)

ILE Processing

8.1.1 Application Configuration

Three new fields have been added to application configuration:-

- ILE Processing Enabled: The basic on/off flag to tell SEE/Change to cater for ILE. The most visible change will be the new options available in the development manager, explained later in this document (See Work with Part using SCDM).
- ILE Remote Delivery Method: Currently SEE/Change delivers program objects. This can be changed for ILE to distribute only certain modules and bind them to their parent programs on delivery. This method of delivery refers only to remote sites and affects the contents of a release packet. All promotes on the development system will act on the program objects. Valid values are; MOD for module distribution and PGM for program distribution. For further notes on this subject see section entitled *Bind Capability*.
- ILE Module Pool Library: The 'module pool' is the name given to the 'live' module object library. On a promote to live at the development centre, module objects will be delivered to this library. The library name can be the same as the live program library if required. Site specific modules will be delivered to their respective site specific libraries. Module objects are kept for re-use by programmers.

```
THNDEV
                                SEE/Change Testing Environment
. . . . . . . . . .
             Application Configuration Extended Options
   Document Processing Enabled . . . Y
                                                                (Y)es/(N)o
  Document Top Level Folder . . . : T#
ILE Processing Enabled. . . . . Y
  ILE Processing Enabled. . . . . Y (Y)es/(N)o)
ILE Remote Delivery Method. . . . MOD PGM/MOD
ILE Module Pool Library . . . . AP1MODS____
                                                                                      : IB cmd dft)
:
                                                                                       : : *NONE
: F3=Exit F9=Command F12=Cancel
        .....
Distribute source code ? . . . : N (Y)es/(N)o
Distribute object override info ?: Y (Y)es/(N)o
Message file operations . . . . : D
                                                      (M)erge/(D)uplicate
Multiple versioning ?. . . .
                                          . : Y
                                                      (Y)es/(N)o
Planned concurrent development ? : Y (Y)es/(N)o
Auto revert if promote errors ? : Y (Y)es/(N)o/(P)rompt
Number of archiving levels . . : 03 0-99 (0 = no Archiving for Appl)
Configuration complexity . . . : 3 1=Simple 2=Intermediate 3=Complex
F1=Help F3=Exit F4=Prompt F9=Cmd F12=Cancel F16=Update
F18=Extended Options F24=Messages
```

8.1.2 Changes to Work with Parts using SCDM

The most extensive changes relating to ILE are within the SCDM panel. Firstly SEE/Change has always displayed one subfile line to represent a program object and its source. For ILE programs the component parts of an ILE program each have their own separate line. One for the source member, one for the module and one for the program itself. This was done as each part of an ILE program has its own characteristics and manipulating functions. Throughout SEE/Change each of the component parts have a unique attribute known to SEE/Change.

For Example:-

- ILE Source members are shown as RPGLE_SRC, CBLLE_SRC, CLLE_SRC and CLE_SRC.
- ILE Service source (Binder Language) is shown and should be retrieved as SRV_SRC.
- ILE Modules are shown as RPGLE_MOD, CBLLE_MOD, CLLE_MOD and CLE_MOD.
- ILE Programs are shown as RPGLE, CBLLE, CLLE and CLE.
- Service programs are shown as RPGLE_SRV, CBLLE_SRV, CLLE_SRV and CLE_SRV

Therefore when retrieving ILE source into your CR, you must include the suffix _SRC to the type parameter. The following panel shows the new SCDM panel and the entries needed to retrieve RPG source for member RTV01.

	SEI	E/Change Testing Envi:	ronment	
	D	Work with Parts using	SCDM	
File Library Type options, pro 1=Retrieve 7=Freeze	• *ALI • T#10 ess Enter. 2=Edit 8=Display	L 0022001 4=Delete obj 11=Transfer	5=Display 12=Work with	6=Print 13=Change obj
Opt Object T 1RTV01R CLCVAT R CLCVAT R CUSMLE R CUSWLE R CUSWLE R CVTDAT R CVTDAT R CVTDAT R INV600 R	ype 2 PGLE_SRC_ PGLE_MOD PGLE_SRC 0 TAARA 0 PGLE_MOD 0 PGLE_MOD 0 PGLE_MOD 0 PGLE_SRC 0 PGLE_SRC 0 PGLE_SRC 0 PGLE_SRC 0 PGLE_SRC 0	Text Calculate_VAT_routine Calculate_VAT_routine Customer_Data_area_ Customer_Selection_win Convert_Gate_routine_ Convert_date_routine_ Convert_date_routine_ Invoicing_Main_Module	ndow	
				More
F3=Exit F4=Prom F10=Hide modules	pt F5=Refi F11=Chg v	resh F6=New ILE pgm view F12=Cancel F20=	F7=New SRV pgm =More keys F23=1	F9=Command More options

Several new function keys have been added:-

F6=New ILE pgm: Use this key to create a brand new ILE program. (This option replaces the PDM requirement to use option 26 to select modules). Creating an ILE program is described below (See Creating a new ILE/Service Program).

- F7=New SRV pgm: Use this key to create a brand new service program. Creating a service program is described below (See Creating a new ILE/Service Program).
- *F10=Hide modules:* This key has been included to help reduce screen clutter.
- F17=Programmer options: New programmer defined options can be specified. Detailed later in this document (See Programmer Defined Options).

New/Changed subfile options:-

- Option 25=Prompt compile: This option was formerly option 15.
- Option 15=Create Module: This new option will compile ILE source into a module.
- **Note:** The new options and function keys described above are only visible and accessible when ILE is enabled. (*See Work with Application Configuration*).

8.1.3 Creating a new ILE/Service program

The following diagram shows the window that appears when F6=New *ILE Pgm* is pressed, it shows a list of modules that can be selected in order to create a new ILE or service program.

	SEE/ Wo	Change Testing Environment ork with Parts using SCDM	
: : 1=Select	module 2=Se	Create New ILE Program elect module (As Entry Module)	:
: Part : SRV	Type C (C/P)	Description	Level :
: _ SRVMOD1 : _ SRVMOD2	Module Module	Service program Module 1 Service program Module 2	*BAS : *BAS :
:SUNDAY	Module	Sunday Module	*SIT SOT:
BT01	Module	beta test 1	*BAS :
: _ CPFG	Module	Current profile group	*BAS :
: _ CRA1 : _ CRTDLTSH	Module PC Module	CRA performance test Create / Delete user space in	*BAS : *BAS :
: More: : F1=Help F12=Cancel : : :			
:	•••••		 More
F3=Exit F4=F F10=Hide modu	Prompt F5=Rei les F11=Chg	Fresh F6=New ILE pgm F7=New SRV view F12=Cancel F20=More keys	pgm F9=Command F23=More options

The list of is split into two parts. The first part contains all the modules registered in the current CR. The second part of the list (After the heading 'Module Pool Components'), contains the modules in the module pool, ie the live library. Programmers can then select their new/changed module from the CR and bind it to any module available in the live pool. The 'Level' column denotes whether this module is a base application module or site specific.

The 'position to' processing for the subfile is conditioned by the C/P flag after the 'position to' module field. A 'C' denotes the 'position to' will align to a module in the CR portion of the list, a 'P' represents the module pool portion of the list.

To select modules for the new program simply place a '1' against the desired module. The entry module for the program can be selected by using option '2'. Nb. When creating a new service program option '2' is used to select the Binder source which will also be listed.

On pressing enter on this screen the prompt for the CRTPGM command itself is shown, at this point it is possible to set any other parameters for the create such as binder directories and activation groups.

8.1.4 Updating an ILE/Service program

ILE programs are non source based, so you cannot directly retrieve an ILE program using the 1=Retrieve method. Instead an ILE program will be updated as a result of changing a component module. Modules can be changed by retrieving and recompiling the source in the usual manner. The next step is to take option 14 against the module object itself, this would present an update window as shown below.

	SEE/Change Testing Enviro Work with Parts using S	nment CDM
: 1=Update Proc Program Ty	Update Programs gram 2=Retrieve Then Update ppe Description	Module CVTDAT : 3=Retrieve only : Also in : Dev CR :
: 1 CVTTAX * H : 1 INV600 * H : 1 INV610 * H : 1 ORD500 * H : 2 SITTHREE * H :	2GM Convert Taxation 2GM Invoicing Main Module 2GM Invoice Print Main Mod 2GM Order processing Main 2GM Program Not in CR	ule Module
: F1=Help F12= : F3=Exit F4=Prompt F10=Hide modules	=Cancel : F5=Refresh F6=New ILE pgm F F11=Chg view F12=Cancel F20=M	: Bottom: : : More '7=New SRV pgm F9=Command Nore keys F23=More options

This panel shows the result of taking option 14 against the module CVTDAT (Top right hand corner). The list details all the programs that this module is known to be bound to. The default option *1=Update Program* is put against all the programs that are currently in the CR. Program SITTHREE is not found in the CR and so has defaulted to option *2=Retrieve then Update*, the program will be brought in from its live library. Option *3=Retrieve only* will bring the program into the CR but not update it (not a default option).

You have the opportunity to deselect programs if you wish. Pressing enter in the above scenario will update the first four programs via the UPDPGM command, then it will duplicate program SITTHREE into the CR, register it and run an UPDPGM against it with module CVTDAT. In the above example program INV600 is also being developed in CR 100235/06. Concurrent development is explained later in this document (See *Concurrent Development Issues*).

8.1.5 Restructuring an existing ILE/Service program

It may sometimes be necessary to change the structure of an ILE program. For example you may want to make one module obsolete and replace it with another, or you may put a call into an existing module to access a further module which now needs to be incorporated in the program.

You may change the composition of a program by taking option 14 against the program object.

		Append ILE Program	Program INV600	:
1=Seled	ct module 2=S	elect module (As Entry Module)		:
Part	Туре	Description	Leve	1 :
	C (C/P)			:
1 CVTDAT	Module	Existing Program Component	*BAS	:
2 INV6001	4 Module	Existing Program Component	*BAS	:
1 CLCVAT	Module	Existing Program Component	*BAS	:
CUSWLE	Module	Customer Selection window	*BAS	:
_ INV6101	4 Module	Invoice Print Main Module	*BAS	:
_ ORD5001	4 Module	Order processing Main Module	*BAS	:
ORD700	Module	new program	*BAS	:
SITE	Module	Site specific module	*BAS	:
SPRLE	Module	ILE SPR Program	*BAS	:
			Мо	re:
F1=Help	p F12=Cancel			:
				:
				:
			M	ore

The above example is of option 14 being taken against program INV600 (Top right hand corner). The first three lines are the known modules that comprise the program. SEE/Change has also defaulted the current entry module (INV600M). The rest of the list is made of modules in the current CR followed by the module pool. The programmer can select or deselect modules before pressing enter to execute a recompile of program INV600.

8.1.6 Deleting/Transferring ILE CR Parts

When an option is taken to delete or transfer a CR part SEE/Change will look for any related objects in the same CR.

For example if a delete or transfer option were taken against a source member, SEE/Change will offer to delete any compiled module(s) and any programs that the module is related to.

If a delete or transfer option were taken against a program object, no checks are made as relationships are checked forwards only. (Source \rightarrow Module \rightarrow Program).



In the above example a delete request for the source of module CLCVAT (Top right hand corner) was requested. SEE/Change offers to delete the compiled module object and the programs the changed module is in. The programmer can now decide to de-select either the module or program by removing the **1=Delete Part** default. Alternatively pressing **F23=Delete requested part only** will ignore list of relations.

8.1.7 The ILE Relationships

After compiles or updates that change the contents of a CR library, SEE/Change will refresh its ILE relationships.

This method of building the cross reference *after* a compile means that SEE/Change makes no attempt to predetermine the module composition of a new program. The programmer can either enter the modules s/he knows about or use binder directories to resolve module dependencies. After a program has been created SEE/Change will have built a cross reference for the program, therefore for future changes to the program a binder directory should not be necessary. For this reason SEE/Change does not store the binder directory used to create a program.

SEE/Change will refresh its cross references when performing a check CR and records the following information:

- The module(s) created from a source member (Source to Module).
- The program created from a source member (Source to Bound Program).
- The program created from module(s) (Module to Program).
- The entry module of the ILE Program.
- The activation group of the program.

8.1.8 Concurrent Development Issues

Concurrent development of ILE source is handled in exactly the same way as other source based objects within SEE/Change.

ILE Programs however that are not source based, are not restricted by concurrent development.

For example, supposing a program consisting of MODULE 1 and MODULE 2 exists. A programmer retrieves the source for MODULE 1, compiles it and updates the program into his/her CR. If that program were locked as a concurrent development object it would mean that another programmer would not be able to retrieve MODULE 2 into his/her different CR and update the program independently. Therefore simply working on MODULE 1 would lock out MODULE 2. The situation would be further complicated if MODULE 2 were bound into another program as it would create a lock on that program also.

For this reason SEE/Change will not restrict the concurrent development of ILE program objects. It will however always make sure that a programmer is aware that somebody else has the program in another CR. (See *Updating an ILE/Service Program*).

8.1.9 Bind Capability

SEE/Change will keep track of a whether or not a program is *bind capable*.

A program is bind capable, if a version of the program already exists in the live environment on the development machine. Meaning that as the object exists in live, it is not necessary to send the whole program to remote production environment on change of a component module, instead individual modules can be sent and bound to the existing object in the production environment.

A program is not bind capable in the following circumstances:-

- The program is new and does not exist in the development centre's live environment.
- The program has been restructured, ie modules have been added or removed from the program changing its composition.

A bind capable program will appear in the *Work with Parts using SCDM* panel as *Dist: Bind.* If remote distribution method (See *Work with Application Configuration*) is set to modules, these programs are not physically packed into a release packet (See *The Movement Mechanism* (*BDR)).

Summary Notes (4.4300):

- Bind capability only applies to deliveries to a production system. All promotes to all environments on the development system are by complete program object.
- SEE/Change keeps track of bind capability regardless of application configuration (To allow the configuration to be changed).

 Bind capability is set according to the contents of the development systems live environment. If an expected program does not exist in a production system a bind error may occur.

8.1.10 Site Specific Modules (4.4300)

Since modules are source based they can be set up as site specific. However modules are not executable objects so this information needs to be passed on to the program object itself. Since program objects can be made up of more than one module a formula is required:-

A program object will default to *BAS.

If any site specific modules are found in the program it will adopt the site of that module.

Example programs:

Program 1	Program 2	Program 3
Module 1=*BAS	Module 1=*BAS	Module 1=*BAS
Module 2=*BAS	Module 2=*SIT SYD	Module 2=*SIT SYD
Module 3=*BAS	Module 3=*BAS	Module 3=*SIT HKG
Adopts *BAS	Adopts *SIT SYD	Adopts *BAS

8.1.11 Version Numbering for ILE Programs

ILE Programs are non-source based objects. However enhancements have been made to allow ILE programs to have version numbers.

Therefore an ILE program's version number is not connected to the version number of any given source member unless the ILE program is a bound program (As the source to program link still exists).

For example a new program can be created from two modules (A and B), the first module (A) at for example Version 005 and module (B) at version 004. The new ILE program object will still be version 001.

Subsequently if module (A) were retrieved into a CR its version number would be incremented to 006. When an update was issued against the program (And the program is retrieved into the CR) The program version will become version 002.

8.1.12 Development Limitations

Certain limitations have had to be imposed, as follows.

- You cannot create a service program beginning with the letter 'Q'. The IBM compiler binds its own service programs to your programs (IBM programs start with the letter 'Q'). SEE/Change will confuse your service programs with the IBM ones it ignores.
- You cannot use binder directories with bound programs. Bound programs are designed to be created with a single temporary module that is disposed of after compilation. SEE/Change relationship handling may be compromised if a

program mixes techniques. If you wish to bind modules to a bound program, then you should re-compile the bound program as a module.

 A bound program must have the same name as its source. As with current programs the source and object must have the same name. Only with real ILE programs with multiple modules bound into a program can the names vary.

8.1.13 CHKCR - CR Checking

Function CHKCR will refresh the ILE relationships for the given CR before processing checks.

To cater for the new ILE application parts CHKCR has been extended to cover the following possibilities.

- There must be an object present in the CR for any source, ie source must have a module or program object present.
- If a bound program is present (Created with CRTBNDRPG) a source member of the same name must exist.
- A Module object must have at least one related program object.
- Source must not have changed since module or bound program creation.
- Module must not have been changed since it was bound to a program.
- If a program is set to 'bind capable' (See *Bind Capability*) then at least one component module must exist in the CR if the application is configured to distribute modules.

8.1.14 The Movement Mechanism

The movement mechanism has been enhanced to cater for ILE objects. This includes new movement type operations and new error codes for them.

New Movement types:

- On a promote to the live environment modules are moved to the module pool libraries. This movement code is *MPU (Module pool update).
- When packaging a release, if the distribution type is by module and the program is bind capable (See *Bind Capability*) (ie The program object is not being sent, only its related modules are being sent to be bound on arrival). The program movement code will be *BDR (Bind reference only).
- If compile is stipulated for promotes, modules will be compiled as normal with the *CPL (Compile) code. The program will have code *BCM (Bind compiled modules). Since ILE programs consist of two or three part records (See Work with parts using SCDM) if a source member is compiled as a bound program, then there is no need to process the program object record. Similarly if a module's source is compiled directly into a module pool library there is no need to perform the module pool update movement for the module record. This situation

could come about if the module were site specific or the module pool library is the same as the live program library. In these circumstances the superflous object record will have the movement code *PCO (Previously compiled). and will not be processed.

 If compile for promotes is not stipulated but modules are to be bound on arrival, the program object will have a movement code *BND (Bind).
 If the program is not found in the target library SEE/Change will get the latest version from the live environment, this code is *BDL.

Summary:

Code	Description	Further Comments
*MPU	Module Pool Update	On movement to live only
*BDR	Bind reference	Program object not packed, record is reference only
*BCM	Bind compiled modules	Prior module compilations have been bound into program
*BND	Bind	Modules have been bound into program
*BDL	Bind: program duped from live	The program was not found in the target library. The program was copied from live to the target library before binding modules.
*PCO	Previously compiled	Movement skipped

New Error Codes:

Code	Description	Further Comments
*E70	Bind parameter error	The bind on arrival program did not receive a complete set of parameters. This should never occur as it indicates a configuration error.
*E71	No modules found to bind	Program was set to bind on arrival but no modules were found that related to the program.
*E72	Update program command failure	The command that performs the module bind to related program encountered an error.
*E73	Duplicated program from live for bind failure	If bind on arrival to acceptance, if the program is not found in acceptance libraries, the mechanism will attempt to copy the program in from live before binding in the new module(s). This indicates a failure to duplicate from live.

8.2 Programmer Defined Options (4.3000)

A new facility for programmers to create their own commands has been included in the *Work with Parts using SCDM* panel.

Pressing *F17=Programmer Options* will present the following panel:

	Programmer options
	Type options, press Enter.
	2=Change 3=Copy 4=Delete 5=Display
	Opt Option Command
	CB CALL BIL500 IRNO(&13) SEQN(&14) APPL(&10)
	_ CC CALL &2/&1
	_ DJ DSPJOBLOG
	DM DSPMSG
	EL EDTLIBL
	_ SD STRDBG (&2/&1) UPDPROD(*YES)
	_ SP WRKSPLF
	_ WA WRKACTJOB
	_ WP WRKSPLF
More	
F12=Cancel	F1=Help F3=Exit F5=Refresh F6=Add F9=Command

Certain values can be used within your own command string. The following diagram shows the panel presented when creating a new option.



The parameters available for substitution in programmer defined options are as follows:

- Object Name
- Object Type
- Object Attribute
- File Name
- Member Name
- Library Name
- Application ID
- System ID
- IR Number
- CR Number

8.3 IMPORT/IMPORTLIB Upgrade (4.3001)

The commands IMPORT and IMPORTLIB have been upgraded to cater for ILE objects.

The IMPORTLIB command can be used to import an external library containing ILE programs, modules and source into a CR library. The CR must have been created with ILE source files configured (ie created since upgrading to 4.3000). For more information about IMPORTLIB see the Development Manager user and reference manual.

Once the import has been executed it is necessary to run a Check CR (CHKCR *TST). It is at this point that SEE/Change will register the links between source, modules and their programs. Once the links are built normal development can commence in the CR.

8.4 Physical file Triggers (4.3002)

An unexpected authority problem (at V3R1) has occurred in the physical file trigger processing program.

The problem manifests itself by not allowing physical files to be compiled more than once in a CR.

Until such time that a full solution to the problem can be provided we have had to disable the physical file trigger processing program.

8.5 Delete application part (4.3003)

A new facility to enable the deletion of object from application libraries. Taking option 23 against a CR object will mark it for deletion. The following panel shows the changes to the WRKCROBJ panel:

	SYl Test Developmen Work with Parts usi	t System .ng SCDM	
File Library Type options, press Ente 14=Compile 15=Crt M 19=Document 20=Mover	*ALL #10025213 er. Module 16=Execute ments 21=History	17=SDA/RLU 18=Scan DB Rel 22=Overrides 23=Mark Delete	
Opt Object Type	Text		
FX416 RPG IIM PF	Foreign_exchange Item Master		
Z3 MKDL1 CLP MX415 RPG	Debenture MX_finance_calculat	Marked Delete	
		Bottom	
F3=Exit F4=Prompt F5= F10=Hide modules F11=C	Refresh F6=New ILE pg ng view F12=Cancel F	m F7=New SRV pgm F9=Command '20=More keys F23=More options	

Objects marked for deletion are processed only upon promotion to the live environment. Therefore a database file will be deleted from both module and acceptance environments when the delete is promoted to live.

You cannot create a new program with the same name as a previously deleted one. This is to prevent object histories being inaccurate and to enable the future enhancement of source conversions to be implemented. (A source conversion for example entails RPG38 to RPG or RPG to RPGLE where you would want to keep history information intact).

Should you decide to mark an ILE object (4.3003 only) for deletion SEE/Change will check for dependent objects and offer to mark any related items. ie deleting a module may affect several programs that contain it therefore the programs are also given, as in the next figure:

		SY1 Test Develop Work with Parts	ment System using SCDM		
					••
:	Mark F	Related Parts for	Deletion	Request: TB001	:
: 1=Mark Par	t for Deletic	n		Type . : *MODULE	:
: Part	Туре	Description			:
: 1 RTV01	*PGM	test1			:
:					:
:					:
:					:
:					:
:					
					:
•					:
÷				Dattor	•
	10 01 10		Dawt Onla	BOLLOI	
: ri-neip r	12-Cancel F2	S-Mark Requested	Part Only		·
:					:
:		•••••	• • • • • • • • • • • • • • • •	••••••••	.:
F3=Exit F4=P F10=Hide modu	rompt F5=Ref les F11=Cha	Tresh F6=New ILE	pgm F7=New SF	Botto V pgm F9=Command F23=More options	m

ILE object relations window showing that the current request to mark module TB001 for deletion should include marking program RTV01.

8.6 CR Checking (CHKCR) (4.3003)

CR Checking has been enhanced to cater for deletions. Basic integrity checks are made to ensure that you cannot delete a physical file and leave a logical view over it. Warnings are issued if ILE dependencies are violated.

The program will not check any program dependencies on deleted database objects.

8.7 Support for ILE SQL (4.3004)

SQL programs written for the ILE are now supported for COBOL, RPG and C. Due to the ever expanding object attribute names the source attribute names for COBOL and RPG have been truncated to SQLCBLLE_S and SQLRPGLE_S respectively to keep within the IBM ten character name convention. Once compiled a module has an attribute of RPGLE_MOD or CBLLE_MOD and not SQLRPGLE_M etc in keeping with the IBM method.

8.8 Work with Parts using SCDM (4.3004)

This program has undergone enhancements to its subfile processing. The Type field used for sorting the list now subsets the list. This makes the recently introduce F10=Hide modules key redundant and so has been removed.

8.9 Source Conversions (CVTCRSRC) (4.3004)

A new facility to enable the conversion of a source member registered within SEE/Change to a different type. For example to convert an RPG38 source to RPG or RPG to RPGLE. Using this function will keep the history details of the source intact.

The following sequence shows an RPG program being converted to RPGLE.

Step 1: Take new option 40 (Convert Source) against the required member

SY1 Test Development Sy	ystem
Work with Parts using S	SCDM
File *ALL Library T#10026501 Type options, press Enter. 24=PDM 25=Pmt Compile 29=Find string 3 47=Exc Cmp/Mrg 48=Cmp/Mrg Rept 49=Mrg Composit 6	30=Batch Find 40=Convert Src 59=Config opt
Opt Object Type Text	
R1PPGM RPGLE test_rpgle_detection	
40 STDRPG RPG Standard_RPG	
	Bottom
F3=Exit F4=Prompt F5=Refresh F6=New ILE pam H	7=New SRV pgm F9=Command
F10=Hide modules F11=Chg view F12=Cancel F20=M	More keys F23=More options

After validation (Concurrent development checks etc) SEE/Change will compose a list of the possible targets for the conversion as follows:

SY1 Test Development System Work with Parts using SCDM Convert CR Source (CVTCRSRC) : : : Source member to be converted : STDRPG Type : RPG : : : Description . . : Standard RPG : : : Select the source type to convert to, press Enter. 1=Select : : Opt Name Attribute : <u>1</u> *PGM RPGLE_SRC : _ *PGM RPG38 : _ *PGM SQLRPG Bottom : • : F1=Help F3=Exit F12=Cancel : F3=Exit F4=Prompt F5=Refresh F6=New ILE pgm F7=New SRV pgm F9=Command F10=Hide modules F11=Chg view F12=Cancel F20=More keys F23=More options

Select the desired type from the list. If the new type is configured to live in a different source file you should ensure that the target source file resides in the CR library.

Selecting a conversion to RPGLE will prompt SEE/Change into offering you the chance to run CVTRPGSRC the IBM supplied command. If you select not to you will have to arrange to convert the source at a another time. The following panel gives an example of the prompt.

SY1 Test Development System Work with Parts using SCDM Convert CR Source (CVTCRSRC) : • Source member to be converted : STDRPG Type : RPG : D : S: Do you wish to run the IBM source conversion utility : 1: (CVTRPGSRC) for this member? : Y (Y/N) : : Enter F1=Help F12=Cancel : : Op :..... 1 *PGM RPGLE_SRC *PGM RPG38 *PGM SQLRPG Bottom : : F1=Help F3=Exit F12=Cancel F3=Exit F4=Prompt F5=Refresh F6=New ILE pgm F7=New SRV pgm F9=Command F10=Hide modules F11=Chg view F12=Cancel F20=More keys F23=More options

The conversion process will stamp your current source as (Marked Delete) and a new record will appear with a retrieval type of *CVT (Convert), as shown in the panel below.

SY1 Test Development System Work with Parts using SCDM										
<pre>File *ALL Library T#10026501 Type options, press Enter. 24=PDM 25=Pmt Compile 29=Find string 30=Batch Find 40=Convert Src 47=Exc Cmp/Mrg 48=Cmp/Mrg Rept 49=Mrg Composit 69=Config opt</pre>										
Opt Object	Туре	Attr	Level	Ver Status	Additional					
R1PPGM STDRPG STDRPG	*PGM *PGM *PGM	RPGLE RPG RPGLE_SRC	*BAS *BAS *BAS	001 *NEW 10/07/9 007 *DEL 23/07/9 007 *CVT 23/07/9	96 Dist: Obj 96 MarkedDelete 96 Dist: Src					
					Bottom					
F3=Exit F4=Prompt F5=Refresh F6=New ILE pgm F7=New SRV pgm F9=Command F10=Hide modules F11=Chg view F12=Cancel F20=More keys F23=More options										

On promotion to live the old RPG source will be deleted due to deletion processing, the new ILE source will be delivered. If the CR were subsequently brought back from archive the old RPG source would be reinstated and the ILE source will be removed.

8.10 Programmer Defaults enhancement (4.3006)

Via the programmer defaults panel (available in SCDM) a programmer can now elect to protect the text description of objects within the CR. This gives SCDM the same function as available in PDM.

8.11 Data Queue Support (4.3006)

Data queues are now supported within SEE/Change. They can be retrieved into a CR and promoted in a similar fashion as data areas.

8.12 Browse XREF Data (BRWXRFDTA) enhancement (4.4001)

Previously this command was available only within a CR. The Command can now be issued from any command line within SEE/Change. To use this command, simply do not enter an IR number or a CR number on the command. The object browser program will then operate with the *retrieve* option unavailable.

8.13 Bound and service program retrieval (4.4002)

Previously it was not possible to just retrieve an ILE program or service program object directly into a CR. This has proved inconvenient to users wishing to simply add a module to a service program. In the past they would have to retrieve a module source, compile it and then take the option to rebind the module, only at this point would the service program be retrieved. It is now possible to retrieve service programs and normal ILE program directly by issuing a retrieve in the normal way and entering an object type of SRVPGM or RPGLE, CBLLE etc.

Some care is required because it is now possible to retrieve the object and never actually make a change, the object will promote normally and its version number will be incremented.

8.14 Hybrid ILE programs (4.4003)

SEE/Change can in a limited** capacity cater for bound ILE programs (Programs created with CRTBNDxxx), but which also have modules attached via the use of a binder directory. Such programs were previously illegal in SEE/Change and had to be either completely built from ILE modules OR be a bound program. The mixing of the two will be referred to as 'hybrid' programs. As with normal SEE/Change the program produced must be the same name as the entry module source. This is because no module object is actually produced therefore the source to object relationship must be linked by name. This is standard with the way the bound programs and all other source-based objects in SEE/Change are handled.

At 4.4003 only provision for applications that neither compile on promote or use cross referencing are supported.

8.15 Non source based item search (4.4003)

A new feature has been added to the member search panel accessed via F16 on the SCDM panel. Pressing F11 on the member search panel will toggle the panel to show non-source based objects. The following panel is an example of the non-source based object search panel.

SY1 Test Development System Work with Parts using SCDM								
File Library Type options, 1=Retrieve 7=Freeze	press Ent 2=Edit 8=Disp	*ALL T#00006014 er. 4=Del lay obj 10=Edi	ete t JDE	5=Displa 11=Transf	y 6=Print er 12=Work with			
• • • • • • • • • • • • • • • • • • • •	• • • • • • • • • • •	Live/Productic	n Object	Search	••••••			
1=Retrieve 5=Display								
Opt Object	Ref Id (P)	Text			Appl (P) Level JDE			
A1 1	JDEDD	Test DD			JDE Test A *BAS			
A1 1	JDEUDC	Test " Al 1"			JDE Test A *BAS			
DSPF1	JDEUDC	UDC for DSPF1			JDE Test A *BAS			
D1D £I	JDEUDC	UDC test			JDE Test A *BAS			
P55015	JDEDD	Test DD			JDE Test A *BAS			
MYDTA	DTAARA	Test data area	1		JDE Test A *BAS			
0#100	RPGLE	Client service	es		JDE Test A *BAS			
					Bottom			
F1=Help F4=P	rompt F6=	Create F9=Cmd	F11=Src	Based F12	-Cancel F24=Messages			

There is no 'take on' job to register non-source based objects within SEE/Change, rather the above list is built piecemeal as objects are promoted to the live environment.
8.16 Database Management Commands (4.4004)

Database Management Commands is a new feature that allows userdefined commands to be processed when a database file is moved as part of a promotion/reversal within SEE/Change. The commands associated with the file can be used to manipulate database members, referential integrity constraints etc. (Note that SEE/Change does not currently handle trigger programs). The commands can be CL commands or SQL statements.

To access the new feature, a new option 27=DBM Commands has been added to the Work with Parts using SCDM panel. This option is only available for retrieved database objects (i.e. PF, LF etc.).

SY1 Test D Work with	Development System Parts using SCDM
File *ALL Library T#00007240 Type options, press Enter. 24=PDM 25=Pmt Compile 26=C 30=Batch Find 31=Refresh MDL 40=C	bj Browser 27=DBM Commands 29=Find string Convert Src 47=Exc Cmp/Mrg 48=Cmp/Mrg Rpt
Opt Object Type Text	
27 ORDDTL PF Order Deta ORDHDR PF Order Head	ler
	Bottom
F3=Exit F4=Prompt F5=Refresh F6=N F11=Chg view F12=Cancel F13=Repeat	lew ILE pgm F7=New SRV pgm F9=Command : F16=Srch F20=More keys F23=More Opt

Choosing this option will present a new set of *Work With* panels that allow the creation, amendment and deletion of the Database Management Commands as shown below.

SY1 Test Development System Work with Database Management Commands CR 000072 / 40							
Object . : MKPF6Type . : *FILEAttribute . : PFSource type/qualifier . : *BASCommand Types : CR specific							
2=Change 3=Copy 4=Delete 5=Display							
OptSeq Typ CommandIR NoCRStatus10FWD ADDPFM FILE(@L/@F) MBR(ORDDTSP) TEXT('Order d00007210Live/Prod10REV RMVM FILE(@L/@F) MBR(ORDDTSP)00007210Live/Prod10FWD ADDFFM FILE(@L/@F) MBR(ORDDTFR) TEXT('Order d00007240Developmnt20FWD ADDFFCST FILE(@L/@F) TYPE(*REFCST) KEY(ODORNO00007240Developmnt30FWD update @L/@F set ODDEPT = CSDEPT where CSDEP00007240Developmnt10REV RMVPFCST FILE(@L/@F) CST(Order_Detail_to_Head00007240Developmnt20REV RMVM FILE(@L/@F) MBR(ORDDTFR)00007240Developmnt							
Bottom F3=Exit F6=Create F9=Command F12=Cancel F16=Resequence F17=Generic cmds							

SY1 Test Development System						
Work with Database Management Commands						
IR/CR: 000072 / 40 This command will be CR specific Object Name/Type/Att: MKPF6 *FILE PF Source Type/Oual : *PRS						
Command Language CLP CLP.SQL						
Command Type FWD FWD, REV (Forward/Reversal movement) Error fails movement. *YES						
Sequence 20 0 = Append last						
Command (must contain substitution variable @L/@F = library/file) ADDFFCST FILE(@L/@F) TYPE(*REFCST) KEY(ODORNO) CST(Order Detail to Header) PRNFILE(ORDHDP) PRNKEY(OHORNO) DLTRULE(*CASCADE)						
F3=Exit F4=Prompt F9=Command F12=Cancel						

Multiple commands can be associated with each file movement, both for moving the file forward (FWD - i.e. promotion) and for reversal (REV). A sequence number determines the order in which the commands will be run. Commands can be defined as *CR specific*, in which case they will only be run when the current CR is promoted and/or reverted, or as *Generic*, in which case they will be run whenever this file is promoted and/or reverted, regardless of which CR the file is in. Access to generic commands is achieved by toggling *F17=Generic cmds*.

The subfile will display all CR specific commands for the file. The commands that have been created in the current CR are highlighted. These are the only ones that can be worked on while in the current CR.

Each command can be any valid CL command or SQL statement. The only restriction is that it must be associated with the declared file. This is ensured by validating that the command contains the substitution string '@L/@F', where @L is substituted with the database library being promoted to, and @F is substituted with the file name. Prompting is available for CL commands.

When promoting a CR, the FWD commands for any files within the CR are run as part of the CR AFTER processing. When reverting a CR for re-development, the REV commands are run as part of the CR BEFORE processing. For each file, generic commands will be run before commands that are specific to the CR. The commands will be run in ascending sequence number order.

Each command can be flagged so that if an error occurs while running the command, the error either causes the whole movement to end in error or the movement ignores the error and continues processing. The *Movement Logs Listing* (OMS440) details each Database Management Command that was run and any errors that occur.

8.17 Compile directives in SDA and RLU (4.4004)

Previously the use of compile directives (/*CPL* etc) did not work properly if you used SDA or RLU. This was because SDA and RLU both insert their own header information above the directives.

The program that processes these has been changed in order to look past these generated headers and find the compile directives. The directives must still appear above the main body of the code.

8.18 ILE Processing (4.4201)

With PE4.4201, various enhancements and bug fixes have been made to the way SEE/Change handles ILE objects.

The commands CRTPGM, CRTSRVPGM, UPDPGM and UPDSRVPGM have been externalised and are now user-modifiable using *Edit ILE Module/Program Commands* on the extended Configuration Manager screen. (This menu option was previously called *Edit Module Compilation Commands*.)

Previously, immediately after an ILE program was promoted to an environment, its constituent modules were rebound into that program (thus ensuring that if a module had been previously promoted to that environment it would be incorporated into the program).

However, it has been found that because of module imports and exports, there may be an interdependency between modules. For example, the rebind of MODULEA may require that a new version of MODULEB exists in a service program. If this service program exists in the CR but has yet to be promoted into the environment, the rebind of MODULEA fails.

Because of this potential interdependency, the processing of the rebinds has changed. The rebinds are now performed after all the objects have been delivered so that there is no longer an issue of the timing of each object's arrival. This is reflected in the OMS440 report where the rebinds now appear at the end of the report.

Also, all the modules for a program are now rebound to that program in one step instead of module by module. This means that if a rebind fails, the program object remains as it was before the rebind step (i.e. the object will not have some modules re-bound and some not). Note however that when reporting this, the OMS440 report will now show that all of the modules are in error. To ascertain which module(s) caused the rebind error, it will be necessary to examine the job log.

SEE/Change now fully supports site-specific and group-specific ILE objects.

The handling of concurrent development of ILE objects has been improved. In general, the restrictions that prevented the retrieval of ILE objects have been changed to warning messages. Conflicts are prevented by improved error checking in the CHKCR processing.

8.19 Display CR current environment (4.4202)

The *Display CR Parts* screen has been enhanced to show the current environment for the CR. (This screen can be reached by choosing option *8=Display obj* from *Work with Change Requests*.) The current environment shows the last environment that the CR was successfully promoted to - *DEV (Development), *MDL (Module Integration), *ACP (Acceptance/QA) or *LIV (Live/Production). Some SEE/Change screens (e.g. *Work with Change Requests*) display the CR status. The CR status differs from the current environment in that it can also display transitory values (e.g. Ready for Testing, Ready for Release, error conditions etc).



8.20 Display Exported Procedures (DSPEXPPRC) (4.4202)

The introduction of ILE has markedly changed the outlook of most iSeries based CM systems. This section contains details of ILE related changes to the Development Manager of SEE/Change at 4.4201.

8.20.1 Introduction

A procedure is a named callable fragment of code that exists within an ILE module - an ILE module may contain many procedures.

When developing an ILE project, common code may be assigned to a procedure. Programs that require this code can then call the procedure. (This makes maintenance easier, as there is only one version of the code.) To call the procedure, a program must either bind the module containing the procedure or bind a service program that contains that module. The problem is that the developer may know which procedure is to be used, but in SEE/Change, what is actually required is the module or service program that contains the procedure. In order to help resolve the module/service program required, a new command (DSPEXPPRC) has been introduced in PE 4.4202 to display exported procedures.

8.20.2 Display Exported Procedures (DSPEXPPRC)

Display Exported	Procedures (DSPEXPPRC)	
Application Exporting Object Source Type Source Qualifier	*ALL *BAS	Name *ALL, name, generic* *BAS, *GRP, *SIT Character value

The display can be invoked from *Work with Parts using SCDM* by pressing *F8=ILE Prc*. The procedures shown will depend on the setting of the *CR Library List Level* for the current CR. If this is *BAS the modules from the configured module pool and the service programs from the live program library will be scanned. If set to *GRP or *SIT the modules and service programs from the configured live group-specific or site-specific library will be used.

SEE/Change Development Environment Display Exported Procedures		
Application : AP1 Test Application 1 Module/SRVPGM : *ALL	Src Type Src Qual.	: *BAS :
Procedure	Object	Obj Type
copy file sub lib	IFSMOD1	*MODULE
cprpthnam	CPR0001	*MODULE
cprpthnam	IFSSRV1	*SRVPGM
cpyjdef	JDEMOD1	*MODULE
crtdltspc	SPCMOD1	*MODULE
delete_arc_sub	IFSMOD1	*MODULE
delete dir sub	IFSMOD1	*MODULE
delete file	IFSSRV1	*SRVPGM
delete_file_sub	IFSMOD1	*MODULE
display file	IFSSRV1	*SRVPGM
Display_Page	IFSMOD2	*MODULE
Display Page Hex	IFSMOD2	*MODULE
Display_Page_Text	IFSMOD2	*MODULE
		More
F1=Help F3=Exit F9=Cmd F11=Next View F12=Cancel		

8.21 Import process (IMPORTLIB) (4.4202)

8.21.1 Introduction

The import process enables users to load objects and/or source into SEE/Change from an external library. Two commands are involved, IMPORTLIB – which loads the Import Register (XIM) with details of the objects/source to be imported – and IMPORT which performs the actual movement of the objects/source named in the Import Register.

In response to customer feedback several enhancements have been identified with the import process and these have been addressed by reworking the IMPORTLIB processing.

8.21.2 Import External Library (IMPORTLIB)

The IMPORTLIB command has been enhanced to include support for group-specific and site-specific imports.

	Import Exterr	nal Library (IMPORTLIB)	
Import Library Name IR Number CR Sequence Number . Source Type Source Qualifier	· · · · · · · · · · · · · · · · · · ·	Name 000001-999999 01-99 *BAS *BAS, *GRP, *S Character value	IT Ə

The processing of IMPORTLIB has been streamlined, so that its function is simply to populate the Import Register – the current Import Mode parameter (*NONE, *REGISTER, *FULL) has been withdrawn, IMPORTLIB will now function as if this parameter had been set to *NONE.

IMPORTLIB has been updated so that it can now correctly handle ILE objects and objects that contain embedded SQL.

When IMPORTLIB is run interactively, a new screen (*Work with Import Register*) will be displayed after the Import Register has been loaded. From this screen, a function key will enable the actual Import to be run (i.e. a *FULL Import).

Note: The *REGISTER mode where the CR object register (XOB) was populated without loading the corresponding objects/source into the CR has been obsoleted.)

8.21.3 Work with Import Register (WRKIMPREG)

Work with Import Register allows the user to manipulate the contents of the Import Register prior to importing the actual objects/source.

Note: This will be called automatically if IMPORTLIB is run interactively.

	Work	with	Import Register	(WRKIMPREG)
IR Number: CR Sequence Number: Source Type:	· · ·	 	• • • *BAS	000001-999999 01-99 *BAS, *GRP, *SIT
Source Qualifier: .			•	Character value

When the Import Register is loaded, the status of all records will be set to *Ready*. If the external library included objects that are not required to be imported, they can be removed from the register by selecting 4=Delete. If certain objects are not required to be imported at this time, option 3=Hold will set the status to Held; these objects will then be ignored by subsequent imports. The Held status can be reset to Ready via option 6=Release.

SEE/Change Development Environment CR Work With Import Register Src Ty Src Qu						/ 16
3=Hold 4=	Delete 6	5=Release	7=Switch Attribute			
Opt Name	Туре	Attribute	Text		Status	Inc Data
XREF GENXREF XREFLST XREFPRE XREFPRF XREFPRF XREFPRP XREFPRT XREFREF XREFSRV XREFST	*FILE *PGM *PGM *PGM *PGM *PGM *PGM *PGM *SRVPGM *PGM	PF-NOSRC NOSRC RPGLE_MOD RPGLE RPGLE_SRC RPGLE_SRC RPGREF RPGRE_SRC	Object cross reference Generate object cross Cross ref list Cross ref pre-compile Cross ref pro-compile Cross ref properties Cross ref properties Cross ref print Cross ref print Cross ref procedures Cross ref procedures	e refer es es	Ready Ready Ready Ready Ready Ready Ready Ready Ready Ready	Ā
XREFTST	*MODULE	RPGLE_MOD	Cross ref testing		Ready Mor	e
F3=Exit F4=Pr	ompt F9=Cn	nd F11=Alt	View F12=Cancel F13=	Repeat	F16=Run I	mport

Certain items (particularly source with no corresponding object) can exist in SEE/Change with differing attributes (e.g. RPG and RPGREF). Where this is encountered when loading the Import Register, SEE/Change will decide on the attribute to be assigned (usually the CPYREF attribute). This can be amended if required by selecting option 7=Switch Attribute.

If a valid alternative is available, a pop-up window will be displayed allowing the selection of the different attribute.

By default, physical files are imported into SEE/Change without data. This can be overridden by typing Y into the *Inc Data?* field on this display. If selected data will be copied into the file in the CR work library version of the file when the import takes place. This will *NOT*, however, set object level overrides to copy the data onwards through the object promotion path. To set these use option 22 from within the CR following importation.

F11=Alt View will change the display of the Import Register to show the external library and source file (if applicable).

F13=Repeat will copy an option to all of the following records in the list. This is cursor-sensitive – place the cursor on the option to be copied before pressing F13.

F16=Run Import will issue the IMPORT command (a pop-up window will offer the choice of submitting the command to batch). All items for this CR that are marked as *Ready* will be registered to the CR and the objects and source copied into the CR library. If the Import runs successfully, the status of these items in the Import Register will be changed to *Imported*.

8.21.4 Import Object/Source Register (IMPORT)

The current Import Mode parameter (*REGISTER, *FULL) has been withdrawn, IMPORT will now function as if this parameter had been set to *FULL.

 Import Object/Source Register (IMPORT)

 IR Number
 000001-999999

 CR Sequence Number
 01-99

The Import process has been enhanced to allow items to be imported under concurrent development. The same rules that govern the retrieval of an object into a CR are applied to the import process and appropriate warning/error messages are issued. If running interactively, these messages will be displayed on screen; if running in batch they will be printed to a report.

8.22 SQL Support (4.4300)

8.22.1 Handling SQL source in a CR

8.22.1.1 Basic concepts

Each of the SQL-based object types can be retrieved into a CR.

The SQL source can be edited using Option 2=Edit and then run in the CR by typing Option 14=Compile.

Before running the source, a validation routine will [interactively] scan the source to ensure that it contains valid SQL syntax, that the objects being referenced are not qualified with library names and that, for DDL, the object being created will have the same name as the source member.

SEE/Change will then make a temporary copy of the source and amend this so that the primary object being referenced is qualified with the correct library name. If the object already exists in the CR and the source is a CREATE, SEE/Change will also add a DROP statement so that the existing object will be removed. A RUNSQLSTM command will then be issued against the temporary source member.

The RUNSQLSTM will produce a report similar to a compile listing that will list the source that was run and detail any diagnostic and/or error messages.

8.22.1.2 Execution messages

The RUNSQLSTM statements that are executed internally by SEE/Change are implemented via an execution message. If required, this can be tailored using *Edit Compilation Commands* in the Configuration Manager; however, there are some noteworthy points as follows:

- the NAMING parameter on RUNSQLSTM must be set to '*SYS' as this allows support for library lists. (SEE/Change assumes that any secondary objects that are declared in the source will exist in the library list).
- the COMMIT parameter should not be set to '*NONE' for DDL object types as SEE/Change treats all the statements in the source member as one unit of work.

8.22.1.3 ALTER TABLE

Having retrieved the source of an existing SQL table you may choose whether to replace the existing table by crafting a new (or modified) CREATE TABLE source or manipulate the existing table by crafting an ALTER TABLE statement. Clearly each option has pro's and con's but essentially, from a SEE/Change perspective, the CREATE TABLE route will replace the object with movements akin to those seen traditionally for PF's whereas the ALTER TABLE route will run the SQL statement against the existing table.

If you choose Option 2=Edit for an existing SQLTABLE, a pop-up window will offer the choice of entering an ALTER TABLE statement, rather than edit the CREATE TABLE source.

SEE/Change Development Environment Work with Parts using SCDM	
<pre>File *ALL Library T#00002202 Type options, press Enter. 1=Retrieve 2=Edit 4=Delete 5=Display 6=Print 7=F i Opt : Do you want to update this table using ALTER TABLE : rather than edit the CREATE TABLE source? : N (Y/N) : Enter F1=Help F12=Cancel 2 MYTABLE SQLTABLE My Table</pre>	t obj : : : :
	Bottom
F3=Exit F4=Prmpt F5=Rfrsh F6=New ILE pgm F7=New Srvpgm F8=ILE Prc F11=Chg view F12=Cancel F13=Repeat F16=Srch F20=More keys F23=More	F9=Cmd opts

Enter Y(es) to craft an ALTER TABLE source and *N*(o) to use the CREATE TABLE route.

If you choose ALTER TABLE, a new member will be added to source file SQLALTSRC in the CR library. You can then enter the ALTER TABLE statement. If you type 14=Compile against the SQLTABLE, the validation routine will be run, SEE/Change will make a temporary copy of the SQLALTSRC member and amend this to qualify the table with the correct library and will add a CREATE TABLE statement so that a copy of the table from Module/Integration will be created in the CR before the ALTER TABLE statement is run. This source will be run using RUNSQLSTM as mentioned earlier.

If the ALTER TABLE source completes with no errors, SEE/Change will append the ALTER TABLE statement to the original CREATE TABLE source member in the CR. This is the source that will be returned to the source pools when the CR is promoted to Live. This will mean that at any point the source pool source will have a complete history of the table comprising the original CREATE TABLE statement along with all subsequent ALTER TABLE statements that have been run against it. If you would prefer that the source returned to the source pools is a single CREATE TABLE statement that, if run, would create the altered table, you can type Option *41=Gen SQL DDL* against the SQLTABLE. This will issue the *Generate Data Definition Language* API (QSQGNDDL) against the table in the CR and the source produced will replace the CREATE TABLE source in the CR.

SEE/Change Development Environment Work with Parts using SCDM	
File *ALL Library T#00002202 Type options, press Enter. 24=PDM 25=Pmt Compile 26=Obj Browser 27=DBM Commands 29=Find 30=B : Opt : Do you wish to generate the SQL source from the : object in the CR using the Generate DDL API? : Y (Y/N) : Enter F1=Help F12=Cancel	string . /Mrg : : :
	.:
	Bottom
F3=Exit F4=Prmpt F5=Rfrsh F6=New ILE pgm F7=New Srvpgm F8=ILE Prc F11=Chg view F12=Cancel F13=Repeat F16=Srch F20=More keys F23=More	F9=Cmd opts

8.22.1.4 Further ALTER TABLE considerations

ALTER TABLE allows you to drop columns from an existing table. When the CR is promoted, SEE/Change will run the ALTER TABLE source at each environment.

If the CR is later reverted, SEE/Change will calculate a reverse ALTER TABLE statement to undo the changes made. So, if the authored source includes the DROP COLUMN phrase, the reverse statement will restore the dropped column(s) by using the ADD COLUMN phrase. SQL will cause these columns to be added as the last columns in the table, so the resulting table may not be physically identical to the original table in terms of level ID.

Furthermore, data in the dropped columns will be lost when restored as part of the reversion movement.

If it is a requirement at your installation that the table layouts and / or data are preserved after a CR is backed out then you should not use ALTER TABLE to drop columns. Instead, use a modified CREATE TABLE statement.

8.22.1.5 SQL DROP command

The DROP statement is not allowed in this implementation of SQL support in the context of a Change Managed part. That is to say SQL statements commencing as follows are invalid:

DROP TABLE DROP INDEX

```
DROP VIEW
DROP PROCEDURE
DROP FUNCTION
DROP TRIGGER
```

To remove a table, index, view, procedure, function or trigger, retrieve the part into your CR and use option 23=Mark Delete.

8.22.1.6 SQLDML

A SQLDML source member can be retrieved into a CR to allow you to enter a SQL DML script. DML source must not contain any DDL statements.

The entered source may be organised into two sections. One section – headed by a comment of '-- **FORWARD**' will be run at each environment as the CR is promoted forward and also if the source is run in the CR. The other section – headed by a comment of '-- **REVERSE**' will be run at each environment if the CR is reverted for development.

The absence of the comment notations implies to SEE/Change that all statements are of the *forward* type.

When a new SQLDML source member is retrieved SEE/Change provides a template source member to help with crafting your DML script.

You can test the source by running it in the CR, however, any files referenced for update in the source must exist in the CR library. If a file has not been retrieved to the CR, duplicate a copy into the CR library manually beforehand.

To run the SQL in the CR, type Option 14=Compile or 16=Execute. Before running the source, a validation routine will scan the source to ensure that it contains valid SQL syntax, that the objects being referenced are not qualified with schema names and that the source does not contain SQL DDL statements.

When executing the DML source (either in the CR or as part of a CR movement) SEE/Change will make a temporary copy of the source and amend this so that the files being updated are qualified with the correct target library name. A RUNSQLSTM is then issued against this temporary source member.

8.22.1.7 Converting PF and LF source to SQL source

SEE/Change supports the conversion of DDS source type PF and LF to SQLTABLE and SQLINDEX / SQLVIEW. The PF / LF must firstly be compiled in the CR, then, type option 40=Convert Src. If you choose to convert to SQL based source, you will be prompted to confirm that you wish to run the Generate Data Definition Language API (QSQGNDDL) against the object in the CR. After the source has been generated, the original PF / LF will be Marked Delete.

Be aware that the generated SQL source may not produce an identical object to the DDS source. If you examine the newly created source, it may contain diagnostic messages on properties of the original file that could not be converted into SQL source.

8.23 Improved source member retrieval display (4.4301)

The search display for source member retrieval has been improved to offer more information to the user earlier in the retrieval processing. Now, the display will show the user the status of the part prior to a retrieval attempt:

Change Management System Work with Parts using SCDM							
File Library . Type options 1=Retrieve 7=Freeze	, press Ent 2=Edit 8=Disp	*ALL W#0001060 er. lay obj	01 4=Delete 11=Transfer	5=Display 12=Work wit	th	6=1 13=0	Print Change obj
		Live/Pro	oduction Member	Search			
1=Retrieve Opt Member	5=Displ Ref Id (P)	ay Text	6=Print		App: Leve	V43 1	V4.4301 De Status
CST0200S	DSPF	Customer	r Lookup Display	/ File	*BAS	01	Live/Prod
CST0210R	RPG	Customer	r Master Lookup	V2	*BAS	01	Live/Prod
CST0300R	RPG	Ship To	Master Maintena	ance	*BAS	01	Accept/QA
CST0300S	DSPF	Ship To	Master Maintena	ance Display	*BAS	01	Accept/QA
CST0400R	RPG	Ship To	Master Lookup		*BAS	01	Live/Prod
CST0400S	DSPF	Ship To Lookup Display File *BAS 01 Live/Prod					Live/Prod
	RPG	Customer Master Listing *BAS 01 Integr/Tst				Integr/Tst	
CST0900R	RPG	Customer Listing by Add Date *BAS 01 Live/Prod					Live/Prod
-							More
F4=Prompt F	6=Create F	9=Cmd F1	10=Non-Source H	711=Chg View	F23:	=More	e Keys

The view may be changed using *F11=Chg View* to reveal further information about the status of the parts that are already in the development lifecycle elsewhere:

Change Management System Work with Parts using SCDM							
File *ALL Library W#00010601 Type options, press Enter. 1=Retrieve 2=Edit 4=Delete 5=Display 6=Print 7=Freeze 8=Display obj 11=Transfer 12=Work with 13=Change obj							
•••••	Live/Production Member	 r Search					
1=Retrieve 5=Displ	ay 6=Print	App:	V43 V4.4301 De				
Opt Member Ref Id (P)	Src Lib Status	IR No/CR User	More CRs				
CST0200S DSPF CST0210R RPG CST0300R RPG CST0300S DSPF CST0400R RPG CST0400S DSPF CST0400S DSPF CST0800R RPG CST0900R RPG	V43DB01SRC Live/Prod V43DB01SRC Live/Prod V43DB01SRC Accept/QA V43DB01SRC Accept/QA V43DB01SRC Live/Prod V43DB01SRC Live/Prod V43DB01SRC Live/Prod	000006 01 RICHARD 000006 01 RICHARD 000009 01 AARON	Yes More				
F4=Prompt F6=Create F	9=Cmd F10=Non-Source	F11=Chg View F23	=More Keys				

'Yes' in the 'More CR's' column indicates that the listed CR is just one of the CR's that feature the part.

In previous versions of SEE/Change F11 was used to toggle the display for non source based objects. This function has now been reassigned to F10. The changes apply to both the source based and non source based displays.

8.24 Overriding duplication of logical file members (4.4301)

8.24.1 Introduction

When an existing logical file is amended and promoted, SEE/Change's default behaviour is to attempt to preserve the logical file's members, so that the new version of the logical has the same member composition and scope as the original.

With PE 4.4301, this behaviour can be overridden so that existing members are not duplicated to the new version of the logical.

8.24.2 Configuration

The *Work with Object Override Details* screen has been amended for logical files so that it includes a new field – *Keep Members?*

SEE/Change Development Environment Work with Object Override Details					
Enter the movement override specifications for object: Application: AP1 Demo Application Object: O#MDLL1 Type/Attr: *FILE LF Level: *BAS Number of target libraries per object: 1 Locate System/Site(P):					
System/Site	: SY1 Deve : 01	elopment	SI1 SI1-	-Dev CM (De	ev Site)
	Config.	Library	Recom- pile?	Jobd OMSJOBD	Keep Mbrs? Y
Live-Production Lib	AP1DTALIV	*CONFIG			
Mdl-Integration Lib	AP1DTAMDL	*CONFIG			
Load Src to Release Packet ? . Source pool library Source pool file	N AP1SRC ODDSSRC				
	2				Bottom
F1=Help F3=Exit F4=Prompt F9	=Cmd F12=0	Cancel F	23=Dlt d	overrides	F24=Msgs

The default setting for this field is blank, which is the same as being set to 'Y', and as such SEE/Change will continue to behave as it has in previous releases.

If this new field is set to 'Y' or blank, then when a new version of an existing logical is promoted to an environment, SEE/Change will examine the existing version of the logical at that environment and record information about the members in this logical. After the new version of the logical has been created, SEE/Change will use this information to add the members found to the new version.

The result will be that the new version of the logical keeps the same member composition and scope as the original.

If *Keep Members*? is set to 'N', SEE/Change will not attempt to duplicate the members that exist in the original version of the logical to the new version of the logical.

If logical files are recompiled during promotion, the result will be a logical file with one member scoped over all the members in the referenced physical file(s) – i.e. the default setting for Create Logical File (CRTLF).

If logical files are duplicated during promotion, the result will be a logical file with the same member composition as the version of the logical in the CR work library.

If the above is not what is required, the logical file members must be handled explicitly (either at compile time or by an *AFTER process).

8.25 New object type DDMF (4.4301)

An object type of DDMF has been introduced in this release. It is treated the same as other non source based database objects. SEE/Change will deliver the object to each configured but will not change the DDMF definition since it has no means to determine where a given DDM file should point. This must be done manually or by some other automated mechanism such as an *AFTER program.

8.26 Execute CR List (EXCCRLST) Utility Command (4.4304)

A new Execute CR List (EXCCRLST) command has been introduced to SEE/Change in this PE. This new command complements the existing Execute Object List (EXCOBJLST) utility command. Its syntax can be demonstrated as follows:

Type choices, press Enter. Development Centre System IR Number CR Sequence Number Dbject Name Object Type	SYSM IRNBR CRSEQ TGTLIB *LIBL OBJ *ALL OBJTYPE *ALL OBJATTR *ALL LIBTYPE *ALL SRCBASED *BOTH SRCUSAGE *ALL EXEC	Character value 000001-999999 01-99 Name, *LIBL Name, *ALL, generic* Character value, *ALL Character value, *ALL *ALL, *DB, *PGM, *JDE *BOTH, *YES, *NO *ALL, *COMPILE, *CPYREF.
F3=Exit F4=Prompt F5=Refresh F24=More keys	F12=Cancel F13=1	Bottom How to use this display

Given a valid system, IR and CR reference (SYSM, IRNBR and CRSEQ parameters) the CR parts list is inspected. For each part that meets the specified criteria (parameters OBJ, OBJTYP, OBJATTR, LIBTYPE, SRCBASED and SRCUSAGE), the command specified in the EXEC parameter is executed.

The Object Attribute (OBJATTR) criteria parameter corresponds to the SEE/Change type, not the OS/400 type. For example, it is possible to run the command and select DSPF parts only.

&1, &2, &3 and &4 are substitution values provided by the command at run time. The &2 (library) substitution value is influenced by the TGTLIB parameter. When set to *LIBL the value is resolved at run time. Otherwise the specified value is passed.

The command will always complete normally and return the OMS0085 informational message. OMS0085 is formed as follows:

```
nnn execution items completed normally, nnn ended abnormally.
```

Users wishing to detect error conditions arising from the use of this command shall need in retrieve this message, inspect its message data fields and proceed accordingly.

8.27 Update Object Source File / Library (UPDSRCFLIB) (4.4304)

A new Update Object Source File / Library (UPDSRCFLIB) command has been introduced to SEE/Change in this PE. This new command can be used modify the source file and source file library service attributes held against OS/400 objects. The syntax is as follows:

Change Object Source File Lib (CHGSRCFLIB)	
Type choices, press Enter.	
Object Name Name Library Name Object Type Character value New Object Source File Name, *SAME New Object Source File Lib Name, *SAME	
Bottom F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display F24=More keys	

To demonstrate, the following command has been used to yield the following OS/400 service attributes display:

DSPOBJD OBJ(ATASG01LO/NNXTL#) OBJTYPE(*PGM) DETAIL(*SERVICE)

Display Object Description - Service	
	Library 1 of 1
Object	
Library ATASG01LO	
Library ASP device *SYSBAS	
Type *PGM	
Source file QRPGSRC	
Library ATASG01LS	
Member	
Attribute	
User-defined attribute :	
Freed	
Size	
Creation date/time	
Source file date/time :	
System level V5R2M0	
Compiler	
Object control level :	
	More
Press Enter to continue.	
F3=Exit F12=Cancel	
(C) COPYRIGHT IBM CORP. 1980, 2002.	

It shows the object was compiled from the QRPGSRC file in library ATASG01LS.

The CHGSRCFLIB command can be used to change the source file and / or source file library. For instance, to change the source file library, the following command may be used:

```
CHGSRCFLIB OBJ(ATASG01LS/NNXTL#)
OBJTYP(*PGM)
SRCFLIB(ATAPG02SS)
```

Subsequent enquiry against the object description service attributes will reflect the change.

```
Display Object Description - Service
                              Library 1 of 1
ATASG01LO
                    *SYSBAS
Туре....:
                     *PGM
ATAPG02SS
Attribute . . . .
        . . . . . . . . . . . .
NO
                     1255472
Size .
       . . . . . . . . . . . . . . .
Creation date/time . . . . . . . . . . . . . . .
                    29/10/04 13:51:27
V5R2M0
More...
Press Enter to continue.
F3=Exit F12=Cancel
(C) COPYRIGHT IBM CORP. 1980, 2002.
```

The source file is left unchanged since it was unspecified.

Note that every OS/400 object has the source file and source file library attribute irrespective of its type. The command may therefore be used against any object.

The UPDSRCFLIB utility makes use of the QUSROBJD and QLICOBJD APIs.

8.28 SEE/Change Plug-in for WDSc (4.5000)

The SEE/Change server side software has been extended to support a SEE/Change plug-in for the Websphere Development Studio client (WDSc) development environment.

The server side software is installed / upgraded in the normal way.

Aside of a small number of SPRs, as detailed in the next section, SEE/Change is functionally unchanged from the previous release (4.4305).

PE 4.5000 is cumulative of all interim releases since 4.4300.

Full details of how to install and utilise the client side software are available on request.

8.29 File Attributes (4.5002)

8.29.1 Introduction

The manner in which SEE/Change retains file attributes throughout the Change Management lifecycle has been streamlined. This change affects the way physical files (PF), logical files (LF), display files (DSPF) and printer files (PRTF) are handled.

The change applies to source based as well as non-source based instances of these object types. It also applies to user defined object types that represent these IBM predefined object types.

Under the new regime, attributes of files being delivered to target environment libraries may be set based on either a) the part in the originating CR or b) the attributes on the existing live version of the file.

8.29.2 Configuration

The primary behaviour of the new mechanism is influenced by a new application level parameter. The new parameter can be found on the secondary application details panel as follows:

THENON THENON Change Managem	ent System 4.5002
Work with Applica	tion Details
Document Processing Enabled N	(Y)es/(N)o
Document Top Level Folder ACM	
ILE Processing Enabled \overline{Y}	(Y)es/(N)o
ILE Remote Delivery Method *BOTH	*PGMONLY/*MODONLY/*BOTH
ILE Module Pool Library ACMELI	VOBJ
ILE Allow missing modules Y	(Y)es/(N)o
Cross Referencing Utilisation *FULL	*NONE/*DATA/*FULL
JDE Application? N	(Y)es/(N)o
SEE/Change Version for JDE $\overline{00}$	(00 - 99)
Development Prod. System(P).	
Development Prod. Site(P)	
Unlock Source at Dev-Prod System . $\underline{\mathbb{N}}$	
IFS Processing Enabled $\underline{\mathbb{N}}$	IFS Archive Library
IFS CR No. of Archive Levels $\underline{00}$	
CRTCRLIB to create schema? $\underline{\mathbb{N}}$	(Y)es/(N)o
Maintain existing triggers? \underline{Y}	(Y)es/(N)o
Set file attributes from CR? \underline{Y}	(Y)es/(N)o
	Bottom
F1=Help F3=Exit F4=Prompt F9=Cmd F12=	Cancel F16=Update F24=Messages

When set to Yes, SEE/Change replicates the attributes of the file in the CR onto the file being delivered. Note that the attributes of the file in the CR will initially be based on the current live version if the file is not a new file.

When set to No, SEE/Change ignores the CR file attributes and instead preserves the attributes of the existing live version of the file. If the file is new, the attributes from the file in the CR prevail for the first promotion only.

After upgrading to 4.5002, the default setting for this parameter is set to Yes.

8.29.3 Overrides

This behaviour can be further influenced by an object level override:

THENON Change Management System 4.5002					
Work with Object Override Details					
Enter the movement override specifications for object: Application: ACM ACME Order Processing System Object: ITM0900W Type/Attr: *FILE PRTF Level: *BAS Number of target libraries per object: 1 Locate System/Site (P):					
System/Site : D43 Developmen Current target library sequence : 01 Recom- Config. Library pile? Job	(Dev System) bd MEIORD				
Live-Production Lib ACMELIVOBJ <u>*CONFIG</u> Acceptance-QA Lib ACMEACPOBJ <u>*CONFIG</u> Mdl-Integration Lib ACMEMDLOBJ <u>*CONFIG</u>					
Set file attributes from CR? . Y Load Src to Release Packet? . Y Source pool library ACMELIVSRC Source pool file QDDSSRC	More				
F1=Help F3=Exit F4=Prompt F9=Cmd F12=Cancel F23=Dlt over	rrides F24=Msgs				

When blank, the application level setting prevails.

When set to Yes, SEE/Change replicates the attributes from the file in the CR when it is promoted.

When set to No, SEE/Change ignores the CR file attributes and instead replicates the attributes from the existing version of the file in live. If the file is new, the attributes from the file in the CR prevail for the first promotion only.

8.29.4 Attribute Retention in the CM Lifecycle

The following flow diagram describes in summary the development lifecycle in respect of file attribute retention.



When an existing source based file is initially retrieved for development, analysis is performed on the existing file in the configured live library and its attributes noted in work files within the CR work library. When SEE/Change is asked to compile the file, the previously gathered attribute information is used to construct a tailored compilation statement. The procedure is worth noting in detail so as to convey an understanding of how the new regime fits in with other elements of SEE/Change. It is:

- Retrieve the compilation execution statement, as defined in the Configuration Manager;
- Substitution values are made. These include target library, object name, source file library, source file name and member name;
- The target release parameter is added to the compilation string. This is absolute;
- The attribute information held in the CR work library is inspected and compared to the default value on the corresponding file creation command. Attribute parameters having values that differ to the default and are not already specified in the compilation command string are added;
- The source is inspected for compiler directives specified using the SEE/Change /*CPL* syntax. These are added to the compile string. If a parameter specified in this way already exists in the compile string, the value specified here replaces the previously specified value;
- If the prompt option is taken, the compilation string as calculated at this point is made available to the user through the regular command prompter. Amendments may then be made directly.
- The compilation string is executed (batch or interactively as elected by the user, if allowed);
- The resulting object is inspected and information about its attributes recorded into the CR work files once again. This replaces the previous data if applicable.

Following development the file will be promoted to an intermediate testing environment or possibly directly to live. Once the new version of the file has been delivered the Apply File Attributes (APYFATR) command is executed to set the attributes on the target file based on a source version of the file. The source version is determined by the application configuration / override. The following table shows the source version of the file, i.e. library, that is used when replicating file attributes.

	Target Environment				
Set file Attributes from CR? setting	Dev Module	Dev Acceptance	Dev Live/Prod	Prod Acceptance	Prod Live/Prod
Yes	CR	CR	CR	Release Packet	Release Packet
No	Live	Live	Archive	Live	Archive

The APYFATR command is described in more detail in the next section.

8.29.5 Apply File Attributes (APYFATR) command.

The Apply File Attributes (APYFATR) command will set the attributes for a named file in one library (the target library) to be the same as a file of the same name and type in another library (the source library).

APYFATR requires four parameters to be completed:

	Apply File	Attributes	(APYFATR)
Type choices, press Er	nter.		
File Name Target File Library . Source File Library . File Type (OTXXX1)			Character value Character value *DSP, *PRT, *PF, *LF
F3=Exit F4=Prompt F24=More keys	F5=Refresh	F12=Cancel	Bottom F13=How to use this display

The attributes and their meaning are defined in further detail below:

- File Name: Specifies the name of the file that is to have its attributes copied / replaced;
- Target Lib: Specifies the library in which the file that is to have its attributes replaced resides;
- Source Lib: Specifies the library in which the file that is to have its attributes copied resides;
- File Type: Specifies the type of file that is the subject of the command. Values are *PF, *LF, *DSPF or *PRTF.

8.29.6 Consistency of configuration considerations.

It is possible to contrive an apparent inconsistency in behavior by having inconsistent configurations. Consider a CR being promoted from a development centre to a production machine where an attribute value is being changed from 10 to 20. The following scenarios exemplifies how such an inconsistency can occur.

Setup:

	"Use file attributes from CR?"
Setting on Development	No
Setting on Production	Yes

Scenario 1:

- CR raised and an existing file is retrieved. An attribute value is changed from 10 to 20;
- CR is promoted to Acceptance then to Ready/Release;
- CR is allocated to a new Release as *LATEST;
- Release is sent to Live on Production Machine;
- CR is promoted to Live on Development Machine.

After the above steps, the attribute values for the file will be as follows:

	Old Value	New Value
Development	10	10
Production	10	20

This is because the file in the release packet will have been wrapped from the CR library, where the attribute value was 20. When installed on the production machine, the new value of 20 will be retained because the application configuration (or override) setting of "Y" for the *Use file attributes from CR* setting will have been observed and respected.

Scenario 2:

- CR raised and file checked out. An attribute value is changed from value 10 to 20;
- CR is promoted to Acceptance then to Ready/Release;
- CR is allocated to a new Release as *LATEST;
- CR is promoted to Live on Development Machine.
- Release is sent to Live on Production Machine;

In this scenario, the final two steps are reversed. Afterwards, the attribute values for the file will be as follows:

	Old Value	New Value
Development	10	10
Production	10	10

This is because the file in the release packet will have been taken from the Live library where the attribute value will have been 10, since it is it the most recent version. Whilst the processing for the production machine will be the same, it is the release packet that is different.

The above merely demonstrates the importance of application configuration consistency across a network of machines.

It is best practice to ensure that consistency in configuration is achieved and maintained.

8.30 Supported File Attributes (4.5002)

As well as having streamlined the manner in which file attributes are supported, the number of file attributes that are supported by SEE/Change has been reviewed and extended at this release. Support for many new parameters has been added; the list of supported file attributes is now as follows:

- **PF:** EXPDATE, MAXMBRS, ACCPTHSIZ, MAINT, RECOVER, FRCACCPTH, SIZE, ALLOCATE, FRCRATIO, WAITFILE, WAITRCD, SHARE, DLTPCT, REUSEDLT, SRTSEQ, LANGID, CCSID, LVLCHK, NODGRP, PTNKEY
- LF: ACCPTHSIZ, FMTSLR, FRCACCPTH, FRCRATIO, LVLCHK, MAINT, MAXMBRS, RECOVER, SHARE, WAITFILE, WAITRCD
- PRTF: AFPCHARS, ALIGN, BACKMGN, BACKOVL, CDEFNT, CHLVAL, CHRID, COPIES, CORNERSTPL, CPI, CVTLINDTA, DECFMT, DEV, DEVTYPE, DFRWRT, DRAWER, DUPLEX, EDGESTITCH, FIDELITY, FILESEP, FNTCHRSET, FNTRSL, FOLD, FONT, FORMDF, FORMFEED, FORMTYPE, FRONTMGN, FRONTOVL, HOLD, IGCCHRRTT, IGCCPI, IGCDTA, IGCEXNCHR, IGCSOSI, IPDSPASTHR, JUSTIFY, LPI, LVLCHK, MAXRCDS, MULTIUP, OUTBIN, OUTPTY, OUTQ, OVRFLW, PAGDFN, PAGERANGE, PAGESIZE, PAGRTT, PRTQLTY, PRTTXT, REDUCE, RPLUNPRT, SADLSTITCH, SAVE, SCHEDULE, SHARE, SPLFOWN, SPOOL, TBLREFCHR, UOM, USRDFNDTA, USRDFNOBJ, USRDFNOPT, USRDTA, USRRSCLIBL, WAITFILE
- **DSPF:** CHRID, DECFMT, DEV, DFRWRT, DTAQ, ENHDSP, IGCDTA, IGCEXNCHR, LVLCHK, MAXDEV, RSTDSP, SFLENDTXT, SHARE, WAITFILE, WAITRCD

The list of supported attributes has been aligned to V5R1. However, no new attributes have been introduced since V5R1 on either V5R2 or V5R3.

8.31 Ability to view Archived Source Members (4.5002)

Previously it has not been possible to view historic versions of source code after the CR has been promoted to live if the application was configured to use an archive library. This was because the CR library is saved to a save file inside the archive library. If such a request was made, the user was informed that the CR library no longer existed on the system, and was invited to view the current live source instead with a popup like the one below:

```
: CR library does not exist. :
Do you want to view current live source ? ..... : Y (Y/N) :
Enter F1=Help F12=Cancel :
```

This processing has now been enhanced to allow the user to choose which source they wish to view, subject to the CR library save file still being available in the archive library.

```
:

: CR library does not exist.

: View Archived source (Y) or Live Source (N) ? .... : Y (Y/N)

: Enter F1=Help F12=Cancel

:
```

If the archived source version is chosen, the source member alone is restored to QTEMP and SEU is launched in browse mode. Depending on system resources and the size of the member that is to be restored, a small delay may occur whilst the restore is being carried out. When the user has finished viewing the source the temporary source file is removed from the QTEMP library. The default is to view the archived source.

8.32 Stream File Support (4.5003)

8.32.1 Folder Retrieval

You can retrieve a folder (or individual stream files or subdirectories contained within it) using the 5250 Development Manager or the WDSC client. Retrieved objects are copied from the register (or optionally from the acceptance or module/integration and acceptance locations if appropriate) to a CR-specific location in the root part of the IFS; the name of the super directory is controlled by the *@IFP* general parameter. You can retrieve one, some or all of an application's folders into a CR. These folder copies are referred to as the *CR work folders*. However the retrieval process does not lock objects against concurrent development; instead checkout is unrestricted and CCD conflicts are resolved either during development or later when it is complete. This CVS-style checkout is appropriate for stream file applications (e.g. a website) that rely on a fixed run-time directory structure, all of which must be present in order to undertake meaningful development and unit testing activities.

8.32.1.1 Retrieval in the WDSc

In the SEE/Change subsystem within Remote Systems Explorer (RSE), the application registers includes a *Stream Files Folders* handle. Expand this to reveal the application's registered folders. Each folder can be expanded to see the content of the folder and, if it contains subfolders, they too can be expanded.

The context sensitive menu includes a *Retrieve stream files* action that may be executed against the entire folder or a single file or subdirectory within the folder. The snapshot below shows retrieval of the entire folder being initiated:



When selected, the *Retrieve Stream Files Folder* wizard is launched and we are asked to select the CR that we wish to retrieve to, as shown below:



Once *Finish* is clicked SEE/Change validates the request and a retrieval dialog may follow, indicated as follows:

💠 Retrieve Stream File Folder	×
Folder //ACMEDEV/www_liv in application ACM The server requests dialog. Please click Next.	
Target CR 001033/01 : Website: Server-side file changes 001033/02 : Website: cgi & HTML changes	
< <u>B</u> ack Next > Einish	Cancel

The most common cause of further dialog is when a folder or subfolder that is not empty is being retrieved. SEE/Change wants to know if you wish to retrieve just the folder, or the folder and all of its content. Clicking Next commences the dialogue:

Retrieve Stream File Folder	
older //ACMEDE¥/www_liv in application ACM Please review the dialog below.	H
-Server retrieve directory dialog	
Do you also want to retrieve any files and subdirectories	
• Yes	
C No	

This feature is useful if a folder is large and the primary intention is to create a new stream file within it, as might be the case for an ftp download site for example.

Once the dialog is complete the Finish button is enabled and the retrieval is submitted to batch. The iSeries Commands Log reports the submission of the request and later reports its completion status:

🗟 iSeries Commands Log	• ×
ACMEDEV	
SBMJOB CMD(RUNRTVSTMF IRNO(001033) SEQN(02) OBJN('//ACMEDEV/www_liv') OBJT('*DIR') FOLDERCODE(0000061) OVRENV('*NONE') INCLUDE('*YES')) JOBD(Y#00103302/CRJOBD) INLLIBL(*JOBD) PRTDEV(*JOBD) PRTTXT(*JOBD) RTGDTA(*JOBD) OUTQ(*JOBD) JOB(RTVCRSTMF) Job 157357/PAUL/RTVCRSTMF submitted to job queue QBATCH in library QGPL.	4
RUNRTVSTMF IRNO(001033) SEQN(02) OBJN('//ACMEDEV/www_liv') OBJT('*DIR') FOLDERCODE(0000061) OVRENV('*NONE') INCLUDE('*YES') Retrieval to CR 001033/02 completed normally Directory //ACMEDEV/www_liv was successfully retrieved to CR 001033/02. 79 objects were copied. Command completed successfully.	4
Command Normal	Jun
Remote System Details Tasks iSeries Table View iSeries Commands Log	

If an error occurs, the iSeries Commands Log will return useful information. If more details are required, then the Error Log (not displayed here) may contain further helpful information.

The folder is now available for development activities within the CR. You can see this by refreshing the RSE display with F5 else right clicking on one of the higher branches and selecting the Refresh function. The CR now appears expandable, as does the folder handle within it along with any subfolders that may also have been retrieved. The following snapshot demonstrates this:



8.32.1.2 Retrieval in 5250

Retrieval is also possible via the 5250 interface. To carry out the retrieval illustrated above, access the CR:

ACME Change Management System Work with Parts using SCDM						
File Library Type options, 1=Retrieve 7=Freeze	 press Ent 2=Edit 8=Disp	*ALL Y#001033 er. blay obj	302 4=Delete 11=Transfer	5=Display 12=Work with	6=Print 13=Change obj	
Opt Object	Туре	Text				
No Parts exist for CR 001033/02 in system D43. F17=Pgmr optns F19=Stream files F20=Keys F23=More options F21=Text F22=Status F24=Messages						

From within the CR option *F19=Stream files* switches to the stream file processing view. The option can be seen by using *F20=More Keys*:

ACME Change Management System Work with Stream File Objects							
CR : 001033 / 02 Website: cgi & HTML changes Application: ACM ACME Order Processing System							
Type options, press Enter. 1=Retrieve 2=Edit 4=Delete 5=View 7=Rtv details 20=Mover	nents 21=History						
Opt Object	Type Stat						
F3=Exit F5=Refresh F9=Cmd F12=Cncl F16=Retrieval Srch No stream file objects exist for this CR.	n F22=Full Path						

F16=Retrieval Search access the list of folders available for retrieval in this application:



From here option *1=Retrieve* will initiate retrieval of the entire folder else option 5 allows navigation into the folder to view its subdirectories.

When selected for retrieval a dialog is initiated with the user in the same way that it is in the WDSc environment:



Eventually the retrieval job is run and, when complete, the folder will appear in the Work with Stream File Objects display:



8.32.1.3 The CR Work Folder

When retrieval has taken place, SEE/Change will have constructed a like-for-like directory structure in a CR-specific work area within the iSeries' IFS. All CR work directories are contained within a special IFS root directory the name of which is:

ххх#ууу

where *xxx* is the SEE/Change work folder prefix value held in general parameter @IFP and *yyy* is the name of the development centre. The shipped value for @IFP is SEE so, assuming that your development centre is called DEV and the value of @IFP has not been changed, a directory called SEE#DEV will have been created.

Within this directory a subdirectory is created for each CR into which stream files or stream file folders have been retrieved. The format for these directory names is:

CRiiiiiicc

where *iiiiii* is the IR number and cc is the CR sequence number.

Within this directory a subdirectory is created for each registered folder or part thereof that is retrieved. The format for these directory names is:

Pnnnnnn

where *nnnnnn* is SEE/Change's internal folder number.

Within here, the original directory structure is reconstructed with the file or files selected retrieved into it. So, in our ACME scenario, the following structure will have been constructed in the root of ACMEDEV:

```
/
+--SEE#DEV
+--CR00103302
+--P0000061
+--cgi-bin
+--public_html etc...
```

In this scenario, the folder happens to be the 61st folder registered within this development system.

If a drive were to be mapped over the iSeries' IFS root, or possibly over a specific CR folder's subdirectory (given a network share), an IDE could be pointed directly at the retrieved version of the stream files.

If you are running multiple SEE/Change environments within a single iSeries partition, for instance a test environment or logical production environments, @IFP must be unique for each one.

8.32.2 The Development Process

If you are using the WDSc you can invoke the LPEX and other editors directly on the retrieved files, which are exhibited in the SEE/Change explorer window. Alternatively you can *export* the folder into a WDSc project and select a new location for the project, which may be a local PC drive or another file system visible in its network neighbourhood. A corresponding *import* action copies the project contents back to the CR work folder. At any stage you can invoke a *show changes* action that provides a tabular listing of the changed (i.e. new, changed or deleted) objects and, once identified, changed parts are *iconised* in the explorer tree view with an information indicator

If the registered folder contained a WDSc project (i.e. if a *.project* file was registered) then the export action creates a new project using that .project file. In this way the export action creates a local renamed copy of a registered project, inheriting the characteristics of the registered project. If the registered folder did not contain a .project file, the export action creates a *simple* WDSC project (i.e. one with no specific nature).

If you are using the 5250 Development Manager you can invoke the EDTF command over any source file in a work folder, and observe its status (*NEW, *CHG or *DEL) change dynamically. However if you want to copy the work folder to some other file system location you must use Operations Navigator to mark the folder as shared and then map a network drive letter over the share.

8.32.2.1 Editing in WDSc

From the RSE tree view select the part that you wish to edit with a rightclick. On the context menu that appears, select *Open With* and a further sub-menu will appear. From here select the editor that you wish to use:

🌆 Remote Systems	20		£	×	**	•	×	
🕀 🔊 New Connection								
🗄 📜 Local								
🕀 👬 iSeries Objects								
⊡ ∰ iSeries Jobs								
IFS Files								
E SEE/Change								
	gement Syst	tem						
	15 Di i Wahaitai		ida fil					
	DI; Website: D2: Website:	; berver-s , cai 0. LIT	ML ch:		iges			
	JZ; WEDSILE; File Objects	; cyra ni		anges				
	MEDEV (MARANA	N live ACM	AF e-c	omme	rce ia	ieh si	ite	
	ai-bin	*_II¥, HCI		omme	100 m			
	oublic btml							
	features							
	images							
T T	📄 pdf							
	acme.css	5						
	Company	/.htm						
	contact.	htm	<u>G</u> o To	2			<u> </u>	
	🖹 demo.hti	m 📃	Орег	With				🛛 🖉 Page Designer
	🖹 downloa	d.htm	Brow	se Wil	th		•	Sy rage besigner
	🖹 featured	l_proi —	-					Web Browser
	📄 index.hti	ml 😽	<u>R</u> efre	esh				Source Editor
	📋 news.htr	m	Expa	nd			1	
	products	.htm	⊆olla	pse				
	this_mor	nth.ht					-	
Applications		-	Rena	me				
		30	Delet	e				
±∎i⊋ iseries Spool Files							_	
		L.	Re-re	e <u>t</u> rieve	€			
			Delet	e				
			Show	CR m	noven	nents	;	
			Show	obje	ct his	tory		
Denishe Cushenia Taran			Prope	erties			-	
Remote Systems Team			-		_	_		

When editing is complete and the changes are saved, they will have been saved directly into the CR work directory, replacing the old version. SEE/Change dynamically spots the change and marks the part with an icon when the tree view is next refreshed:



Note that changes made to files that have been exported to a project won't be flagged as changed until the folder has been imported, at which point changes can be calculated and the tree-view will reflect this.
8.32.2.2 Editing in 5250

From the Work with Stream File Objects display, navigate to a file that is eligible for editing (i.e. not a binary or subdirectory):

ACME Change Management System Work with Stream File Objects CR . . . : 001033 / 02 Website: cgi & HTML changes Application: ACM ACME Order Processing System Directory : //ACMEDEV/www liv/public html Type options, press Enter. 1=Retrieve 2=Edit 4=Delete 5=View 7=Rtv details 20=Movements 21=History Opt Object Type Stat features D ___ images D ___ pdf D acme.css F 2 company.htm F ___ contact.htm F ____ demo.htm F ___ download.htm F featured_product.htm index.html F F More F3=Exit F5=Refresh F9=Cmd F12=Cncl F16=Retrieval Srch F22=Full Path

When selected with option 2=Edit the EDTF function is launched:



As with the WDSc interface, changes are detected dynamically and indicated in the Work with Stream File Objects display:

ACME Change Management System Work with Stream File Objects	
CR : 001033 / 02 Website: cgi & HTML changes Application: ACM ACME Order Processing System Directory : //ACMEDEV/www_liv/public_html	
Type options, press Enter. 1=Retrieve 2=Edit 4=Delete 5=View 7=Rtv details 20=Movements	21=History
<pre>Opt Object features images pdf acme.css company.htm contact.htm demo.htm download.htm featured_product.htm index.html</pre>	Type Stat D D F F F F F F F F F
F3=Exit F5=Refresh F9=Cmd F12=Cncl F16=Retrieval Srch F2	2=Full Path

8.32.2.3 Exporting to a Project

To export a folder to a project, select the entire folder handle with a right click. The context menu offers an *Export to project* option:



Selecting this launches the Export wizard:

Destinat N <u>a</u> me	ion project \$ee.dev.00103302	2.p0000061		Browse
Mode			100m	
🖲 Aļļ f	iles			
	erseded files			
Source -				
€⊆R	folder			
○ *L <u>I</u> '	/ archive			

By default the wizard proposes to create a uniquely named project for you. It's name is derived from the CR work folder path name. If you wish to use an existing project, use the Browse button and select it from the list. Once complete, click Finish to commence the export.

Note that if selecting an existing project, SEE/Change disallows the export if another CR / Folder has been exported to it before. If you wish to reuse a project in this way it will be necessary to delete the project in its entirety and recreate it. If the last export was from the same CR / Folder, then a warning is given.

If the retrieved folder was not a project, that is to say is missing the necessary *.project* file, SEE/Change will construct a simple project into which the retrieved folder's content will be placed.

If the retrieved folder was a project, the newly created, or overwritten project, will assume the attributes of the project that has been retrieved.

If the default option is taken so that a new project is created, SEE/Change will need to know where the project is to be located. A folder destination popup will be provided and by default SEE/Change will select the configured WDSc workspace:

Browse For Folder	? X
Please select location for project see.dev.00103302.p0000061	
Andrew	-
🕀 💼 💼 Default User	
🕀 💼 🛄 Nigel	
🕀 🔁 Paul	
庄 🔂 .eclipse	
🕀 🛄 Application Data	
Cookies	
- Desktop	
🗄 ⊡ Favorites	
🕀 💼 🛄 Local Settings	
🖾 🖾 My Documents	
🗄 🛄 Adobe	
Ė∾ 🛄 IBM	
🕀 🛄 Client Access	
e wdsc	
My eBooks	-
Folder: workspace	
OK Cancel <u>N</u> ew Fo	lder

Select a different location if required and press OK to continue with the export.

Once the export has completed, the Tasks window will show an uncompleted task for the development activity that must follow.

8.32.2.4 Importing from a project

Once your development is complete, the import function is used to repopulate the CR working directory. To do this, select the entire folder handle with a right click. The context menu offers an *Import from project* option:



Select this to launch the Import wizard:

🕀 Import	t Stream Files to 194.205.242.98	×
Into strea Please clic	am file folder //ACMEDEV/www_liv in CR 001033/02 ck Finish to import or Cancel.	
- Source p	project	
Name	see.d43.00103302.p0000061	
	EinishCancel	

By default the import wizard remembers the project to which the folder was exported. If you wish to import a different project's content for some reason, use the Browse button and choose one from the list, but be aware that the existing CR work folder's content will be erased and replaced with that of the selected project. Once complete, click Finish to commence the export.

SEE/Change will remove all of the existing CR work folder contents and repopulate it with the current project content. Once complete the CR work folder will appear identical to the project folder and SEE/Change will have calculated the net change between what was retrieved and what is now in the CR work folder.

8.32.2.5 Viewing Changed Stream files

For a given folder in a CR it is possible to obtain a complete list of the stream files that have changed. To do this, select the entire folder handle with a right click. The context menu offers a *Show object changes* option:



When selected, the SEE/Change Table View is opened in the bottom right hand pane of the workbench and a list of changed files constructed and displayed:

III Changed objects in folder [//ACMEDEV/www_liv] in CR 001						×
Object name	Object	Status	Retrieval	Retrieval	Retrieve	Retrieved size
//ACMEDEV/www_liv/.project	*FILE	*NEW	00/00/00	00:00:00		0
//ACMEDEV/www_liv/cgi-bin/file_update.pl	🖹 *FILE	*CHG	28/10/05	14:36:16	PAUL	3056
//ACMEDEV/www_liv/public_html/acme.css	🖹 *FILE	*CHG	28/10/05	14:36:51	PAUL	6128
//ACMEDEV/www_liv/public_html/featured_product.htm	🖹 *FILE	*CHG	28/10/05	14:36:53	PAUL	6084
//ACMEDEV/www_liv/public_html/images/Offer.jpg	🖹 *FILE	*NEW	00/00/00	00:00:00		0
//ACMEDEV/www_liv/public_html/images/scientiest_big.jpg	FILE *	*DEL	28/10/05	14:36:34	PAUL	8897
//ACMEDEV/www_liv/public_html/this_month.htm	FILE *	*CHG	28/10/05	14:36:55	PAUL	6084
Demote Sustem Datails, Tacks, iSeries Table View, iSeries Com	nande I og I G	EE (Chape	e Table View			4

For each changed object the following information is shown:

- Object name (including full folder path)
- Object type (*FILE or *DIR)
- Status (*NEW, *CHG or *DEL)
- Retrieval date
- Retrieval time
- Retrieval user
- Retrieved size

The horizontal scroll bar allows further information to be displayed about the objects.

8.32.2.6 Deleting Files or Subdirectories

It is important to understand the difference between *deleting* a file or subdirectory from a CR and setting it to being *marked for delete*.

Deleting a file or subdirectory removes it from the CR register and means that it will no longer participate in the CM lifecycle. The file(s) are removed from the CR work folder and as a consequence any changes that were made to the file(s) since retrieval are lost. Use this option if you had not intended to retrieve the file(s) in the first place, or wish to discard changes made. To *delete* a file or subdirectory from a CR, select it within the CR tree view with a right click and choose the *Delete* option.

Marking a file or subdirectory for deletion means that it will be removed from the run-time environment as the CR is promoted. Implicitly, any changes made since retrieval are lost. There is no SEE/Change option to mark a file or subdirectory for deletion; instead, remove it from the CR work directory. When SEE/Change subsequently calculates the net change for the retrieved files, it will determine that the retrieved file no longer exists and will *Mark it for Deletion*. When the CR is promoted it will be removed from the target environment.

8.32.3 Ending the Development Process

Once development activities have been completed and object changes (optionally) reviewed, the CR can move out of development and into the control of the Change Manager. The final stage in this process is to set the CR to a status of *Ready for Testing*.

When you set a CR status to *Ready for Testing*, SEE/Change performs a number of checks to ensure that subsequent promotion is valid. For stream files this checking deals with the possibility that retrieved stream files have become obsolete during the development process. That is to say, a stream file that has been changed in your CR has also been retrieved and changed in another CR and promoted to live ahead of yours. Such files are said to have been *superseded*. If you attempt to set your CR status to *Ready for Testing* when it contains a changed file that has been superseded, you will receive an error message and the CR status will not be changed.

You can resolve these errors (which are analogous to CCD conflicts for library objects) by retrieving the current version of superseded files into the CR work folder. This option is available directly from within the CR, is known as *re-retrieve* and is available as shown below:



This action causes the older version to be renamed so that you will have two versions of the file in your CR working directory.

For example, let's suppose that two users checkout file *acme.css* and both make changes. User 1 promotes his/her CR to live. The file in user 2's CR is now said to have been superseded. When user 2 re-retrieves the file, their changed version is renamed *acme._001_.css*. (The _001_ insertion is based on a file/CR level counter, so it will only advance to _002_ if the same situation occurs again within the same CR). You must decide what to do with the changes you made in the superseded file; you may have to reapply them to the current version or, if your version is absolute and is to replace user 1's file, simply delete the re-retrieved file and rename your original file back to its correct name.

Once conflicts have been resolved the CR can be set to *Ready for Testing* and onward movement initiated.

8.32.4 Further Information on Stream File Processing (4.5003)

This section contains further supplemental information about how SEE/Change handles stream files.

8.32.4.1 Concurrent Development

Concurrent development of stream files is not supported. However, because of the optimistic checkout model used, concurrent development issues only occur when two CRs are responsible for retrieving *and* changing the same file (as opposed to having merely retrieved it and not changed it).

As a consequence of this, several CRs may retrieve a folder and make changes to the files within it. Promotion out of development is permitted for all the CRs so long as no two CRs have changed the same files. As soon as an overlap occurs, the subsequent CR to attempt promotion is denied until either the first CR is reverted back to development or is promoted to live. If the first CR is promoted to live, supersession will have occurred.

To demonstrate this further, consider the following examples:

- Example 1. CR1 & CR2 both retrieve the same folder having FILE1 and FILE2 in it. FILE1 is changed in CR1. FILE2 is changed in CR2. Both CRs may be promoted in any sequence, concurrently or not. This is because although they have retrieved and worked on the same folder, a different subset of the folder files has been changed.
- Example 2: CR1 & CR2 both retrieve the same folder having FILE1 and FILE2 in it. FILE1 is changed in CR1. Both FILE1 and FILE2 are changed in CR2. At this point both CRs may be set to *Ready for Testing* (the second one to do so will receive a warning) but only one of the CRs may be promoted. The other will fail CCD checks and promotion will be denied. If CR1 were promoted to module/integration test and then reverted, CR2 could then be promoted instead. If CR1 were promoted to live, CR2 would be unable to promote at all as supersession will have occurred.

8.32.4.2 Retrieval versus New

If a file is not retrieved and is subsequently created in the CR work directory, a conflict will occur. This is because the CR validation will view the file in the CR work folder as being new, yet it will insist on retrieval because it is already registered. This will be seen when attempting to set the CR to being *Ready for Testing* at which point OME5269 will be issued "New file in CR *xxxxx/yy* has already been registered...".

In order to recover the situation the offending file must be removed from or renamed in the CR working folder then retrieved via SEE/Change. If renamed, then you must decide about any differences between the two versions of the file and rectify accordingly.

8.32.4.3 Develop in Module/Integration

Develop in Module/Integration of stream files is not supported.

Check-in occurs when the first CR movement occurs (i.e. the CR is moved to one of module/integration, acceptance or live). Thus when a CR is promoted to the module/integration test environment, check-in will occur and the CR work folder will be saved and removed from the IFS to prevent erroneous changes being made. Accordingly, access to the stream file functions is prohibited.

8.32.4.4 Re-retrieval

In the earlier example a specific file was re-retrieved.

It is also possible to re-retrieve an entire folder or a subdirectory within a folder. When this option is selected SEE/Change will re-retrieve:

- all files that have been superseded at live irrespective of whether or not they have been changed in the CR and
- all files that have been changed in the CR irrespective of whether or not they have been superseded at live

Consequently this option should be used with care and consideration given to each re-retrieved file before committing the CR for onward promotion.

8.32.4.5 Retrieval of Changed Files at Intermediate Environments

Although concurrent development is not allowed, retrieval of changed files into a CR from intermediate environments is allowed, enabling a degree of concurrency for sequential development activities. Most notably, when changed files exist for the folder being retrieved at either or both intermediate environments, you decide which environment on which to base your retrieved files. Your decision is likely to be based on whether you are making a change that will precede those currently in the intermediate environment (an emergency fix) or follow them.

The following worked examples intend to demonstrate how SEE/Change works in these scenarios.

• A folder with FILEA, FILEB and FILEC has been registered. At the outset all files are assumed to be version 1 (V001).

• The folder is retrieved into CR1 and FILEA is changed. The CR is promoted to the acceptance test environment. The environment will now look like this:



 The folder is retrieved into CR2. During the retrieval process you will be asked whether you wish to retrieve the objects from acceptance (*ACP) or live (*LIV). The dialog appears as follows:

Retrieve Stream File Folder	×
Please review the dialog below.	
Server retrieve from *ACP dialog This object and/or objects within it are also at Accept/QA. © Retrieve from *ACP, © Retrieve from *LIV	
< <u>B</u> ack <u>N</u> ext > Einish	Cancel

- If acceptance (*ACP) is selected, then when retrieval has completed FILEA will be as per CR1's version whilst FILEB and FILEC will be as per the current live versions.
- If *live* (*LIV) is selected, all three retrieved files will be the same as those currently at live.

- In either case SEE/Change will not prevent CR2 from being delivered ahead of CR1 unless FILEA is changed within CR2, since this will result in concurrent development.
- You must decide which is the most appropriate environment from which to retrieve for the purposes of your change. Generally speaking you should select *live* if your fix is an emergency fix that will be delivered before other changes that are currently at acceptance test, and *acceptance* if your fix is to be delivered at the same time or after other changes that are currently at acceptance test.
- We now change FILEB and promote CR2 to the module/integration test environment. The environment now looks like this:



 Finally, we retrieve the folder to CR3. Once again we are asked where we would like to retrieve the files from, but this time we can choose from module/integration, acceptance or live:

and the second se	m File Folder		
older //THENON/ Please review the d	'Folder_L in application	ən ACM	E T
-Server retrieve fr This object and/o	om *MDL or *ACP dialog or objects within it are al:	so at both Accept/QA ar	id Integr/Tst.
Retrieve from	n *MDL, *ACP,		
C Retrieve from	n≁ACP, n*LIV		
1			

- If module/integration (*MDL) is selected, the retrieved files will be as per the module/integration test environment, meaning we shall have V002 of both FILEA and FILEB and V001 of FILEC; ideal if our new change is going to follow those already made by CR1 and CR2.
- If acceptance (*ACP) is selected, the retrieved files will be as per the acceptance test environment, meaning that we shall have V002 of FILEA and V001 file FILEB and FILEC; ideal if our new change is going to follow that made by CR1, but precede that made by CR2.
- If we choose *live* (*LIV), the retrieved files will be as per the live environment, i.e. version 1 of all three files and ideal is our change is an emergency fix that is to be delivered before all other changes.

The scenarios cited here, whilst not exhaustive, should demonstrate that you can base a development environment on live or an intermediate test environment. Whilst this implies a specific delivery sequence SEE/Change does not enforce this unless concurrent development occurs. Because there is no concurrent development in our scenarios (no two CR's change the same file) any of the CR's may be promoted in any sequence, irrespective of which environment the retrieval was based on.

8.32.4.6 Accessing multiple SEE/Change Development Centres

SEE/Change extends RSE's approach of using *filters* to access different parts of an iSeries server. Within the SEE/Change subsystem, a filter refers to a SEE/Change database instance. This comprises three libraries, and these are usually called OMSOBJ (SEE/Change's program objects), OMSDTA (the SEE/Change database) and OMSSAV (the save library that corresponds to the data library), although the client only needs to know about the first two. By default, the SEE/Change plug-in for WDSc is installed with one filter that refers to the default names OMSOBJ and OMSDTA.

If you have multiple SEE/Change development centre databases within a single logical partition / server, then you may wish to be able to access these concurrently from within WDSc. To do this you will need to create a new filter and specify the corresponding data and object library names.

Let's assume that you have a secondary development environment that uses OMSDTA2 and shared SEE/Change's object library OMSOBJ. Right click on the SEE/Change subsystem handle and select <u>New</u> followed by SEE/Change Development System Filter, as shown below:

🌆 Remote Systems	\$ \$		¥ \$ ▼ ×	
🕀 🏄 New Connection)			1
🕀 🖳 Local				
	12			
E Series Obje	cts			
H A iSeries lobs	nanus			
Free IFS Files				
🕀 🚔 All CR's	<u>N</u> ew	P 🍑 SEE/C	Thange Development	t System Filter
🗄 📑 iSeries Spoo	<u>G</u> o Into			
	<u>GO</u> 10			
	🐻 Ogen in New Window			
	E Show in Table			
	🏟 <u>R</u> efresh			
	Expand			
	⊆ollapse			
	Connect			
	Disconnect			
	– Clear Password Cache			
	Propertijes			
Remote Systems Team]

SEE/Change presents the New SEE/Change Development System Filter wizard:

4 New	×
SEE/Change Development System Filter	
Create a new filter to identify a development centre and C	Rs
Object library:	
рмзовј	
Data library:	
OMSDTA	
IR/CR numbers:	
*ALL	
CR sequence	
C IB/CR	
• Priority	
100	
< Park Neith 2	Finish Cancel
< Dark Mext >	

Here we are required to specify the SEE/Change object and data library that comprise our new environment. Seeing as the defaults are OMSOBJ and OMSDTA, we only need to amend the data library to be OMSDTA2.

Note that we may also specify particular CR numbers and the sequence that we wish our CRs to appear within this filter. If you wish to limit the filter to display only particular CRs, specify them in the following format, for instance:

001234/01 001234/02 001235/01

Otherwise leave the default value of *ALL.

Finally click <u>Next</u> to proceed to the next wizard page:

4 New	×			
SEE/Change Development System Filter	_			
😢 Enter filter name				
Filters are saved for easy re-use. Specify a unique name for this filter. This appear in the Remote Systems view, and will be expandable.	name will			
Filter name:	Filter name:			
Only create filter in this connection Select a profile to own the new filter. This determines if it is unique to you, sharable by the team. It will be placed in the default filter pool for that prof	or iile.			
Owner profile: brd01	•			
< <u>B</u> ack <u>N</u> ext > <u>Finish</u>	Cancel			

The second wizard page requires us to specify a name for our new filter. This will be how the filter appears within the SEE/Change Subsystem once it has been created. Specify a filter name, say, *Secondary Development Filter*, then press *Finish* to create it. We are returned to RSE and our new filter will be displayed alongside the original:

📕 Remote Systems	≫ ◆ ◆ 首 X \$ ▼ ×
 New Connection Local Thenon Series Objects Series Jobs SEE/Change All CR's Secondary Development F Series Spool Files 	
Remote Systems Team	

To delete a filter, simply select it with a right click and take the <u>Delete</u> option.

You can change the libraries that a filter uses and its IR/CR selection and sequencing option. To do this select the filter that you wish to change with a right click and select the *Change* option.

It is important to note that the type of filter being discussed here is local to the WDSc workbench and is not the same as a SEE/Change filter that is created and administered in SEE/Change's Configuration Manager and applied by the server.

8.32.4.7 SEE/Change Options / Preferences

SEE/Change allows a number of optional setting to be decided by the user. These options are knows as SEE/Change Parameters and are accessible within the workbench by selecting <u>Window / Preferences</u>. The resulting panel includes a SEE/Change Options entry. Select this to see the options available as shown below:

Preferences		
Preferences Preferences Agent Controller Ant Build Order Data Debug Help Install/Update Internet Iseries Projects J2EE Java LOgging LPEX Editor Plug-In Development Profiling and Logging RAD Remote Systems SEE/Change Options Server Spell Check Symptom Database Team Validation Web and XML Files Web Facing Web Facing Web Services	SEE/Change Options RSE Tree View ✓ Show CR text ✓ Show CR object text ✓ Show Application text ✓ Show Member text ✓ Show Non Source text ✓ Show Folder text ✓ Show Folder text ✓ Show Folder text	Server Data queue wait (seconds): 30 Maximum number children: 500 Stream file CCSID: 1252 Log tree commands Log XML responses Cog prepare commands
Import Export		OK Cancel

8.32.4.8 Error Diagnosis

If you are experiencing difficulties with your configuration, setting the various logging options on will help to provide useful diagnostic information in the error log. There are two sets of logging options to select, both from within the Windows Preferences window.

Select SEE/Change Options and check all three logging options, as shown below:

Log tree commands
Log XML responses
Log prepare commands

Expand the *Remote Systems* preferences and select *Logging*. Set the logging level to *Warnings, errors and informational messages*, as shown below:

Preferences		
	Logging	
🕀 Agent Controller		
🕀 Ant	Specify the logging options for Remote System Explorer	
Build Order		
🕀 Data	Logging Level:	
i Debug		
	C Errors only	
🕀 Install/Update	O Warnings and errors	
Internet	Warpings, errors and information messages	
庄 iSeries Projects	wannings, errors and information messages	
J2EE		
🗄 - Java		
Logging		
庄 - LPEX Editor		
🗄 Plug-In Development		
🗄 Profiling and Logging		
🚊 Remote Systems		
Communications		
File Cache		
Files		
Logging		
Passwords		
Spell Check		
Symptom Database		
±Team		
主 ·· Validation		
庄 ·· WAS Debug		
🕀 Web and XML Files 📃	Restore <u>D</u> efaults	Apply
Import Export	ОК	Cancel

However, you should generally run with these logging levels switched off in order to maintain performance.

9 Release Manager

9.1 Restoring release packet names (4.4000)

A new command has been created to assist in the rerun of failed installs. In some cases the objects in the release packet are renamed. Once renamed the install cannot be run again. Command RSTRLSNAM (Restore Release names) accepts *release packet library name* as a parameter and renames all the objects in the packet back to their original names, thus allowing the install to be run again.

9.2 Overrides for DMS submit RCVRLS (4.4201)

9.2.1 Assigning an Override Id to a Release

When the parameter @ROV, Allow overrides for DMS submit RCVRLS?, is set to *YES at the development centre, a new field is displayed on the *Work with Release Distribution* screen. For each target system, you can define an Override Id (between 1 and 9). When the Release reaches the target system and is about to be installed, the Override Id is matched to the configured *Overrides for DMS Submit RCVRLS* (see above) and any entered override job description and/or schedule date/time will be used when submitting the RCVRLS. If the corresponding override has not been configured at the target system or the parameter @ROV is set to *NO at the target system or the target system is at a pre-4.4201, the DMS submit override Id will be ignored.

```
THENON
                         SEE/Change Development Environment
                           Work with Release Distribution
Release .: 00181 OPEN Test for DMS submit RCVRLS overrides
Target Environment: *LIV Send Type . . . . COMS (Coms/Tape/Intl)
Application . : 1 of 1 AP1 Demo application 1
Type 'X' to select distribution and type any DMS submit RCVRLS overrides.
Target Systems
                                    Dst Ovr
                                               Target Systems
                                                                                    Dst Ovr
SY3-Test Production (* LIVE *) X 3
SY4-Thenon Support Machine
                                          4
                                      Х
                                                                                     Bottom
No more applications are found. Press F21 to instigate transfer.
F1=Help F3=Exit F7=Previous APP F8=Next APP F9=Cmd F12=Cancel F15=Select/Omit CRs F21=Transfer F22=Packaging F24=Msgs
```

9.3 Release Forwarding (4.4305)

9.3.1 Sending a Release from the Development Centre

When sending a release from the development centre, the Release Manager dynamically calculates which systems the release is to be sent to and which are forwarded to from other system. The following display shows how the release will appear on the development centre system in our scenario:

THNSPT Change Management System Work with Release Distribution
Release . : 00097Acme Pricing ModificationsTarget Environment: *LIVSend Type COMS (Coms/Tape/Intl)
Application . : 1 of 1 ACM Acme Sales Order Processing
Type 'X' to select distribution and type any DMS submit RCVRLS overrides. Target Systems Dst Ovr Target Systems Dst Ovr Test System X X X Validation System (* FWD *) _ _ Training System (* FWD *) _ _ Live System (* FWD *) _ _
Bottom
F3=Exit F4=Prompt F7=Previous APP F8=Next APP F9=Cmd F12=Cancel F15=Select/Omit CBs F21=Transfer F22=Packaging F24=Msgs

It shows that each of the other systems are distributed to from elsewhere by using the (* FWD *) notation. When *F21=Transfer* is pressed to initiate the release packet building and sending, the release packet will only be sent to the test system, SY1, unless this has been overridden. When the release arrives at SY1 the release packet is unpacked for local installation and also stored in the release packet archive library, ready for onward distribution.

9.3.2 Forwarding a Release from an Intermediate System

On an intermediate system, SY1 in our case, onward distribution is initiated using the new Release Manager option *16=Forward*:

THNSPT	Test S	ystem			
	Work with	Releases			
2=Change 4 10=Version chk 14 Opt Release Text	1=Delete 5=Wrk 1=Send to acpt 15=Send t	with CRs 8=Status to live <mark>16=Forwarc</mark>	Rls Stat	9=Close 17=Install Rcv-Date	COMS CR Stat
<	Locate release				LAR
16 DEV 00097 ACME	E Pricing Modification		OPEN	1/12/04	Y
DEV 00096 Data	a fix to pricing file		OPEN	1/12/04	Y
DEV 00095 New	lock field in invento	ry master	OPEN	29/11/04	Y
DEV 00084 Stoc	ck transfer calculatio	n correction	OPEN	29/11/04	Y
				Bo	ttom
F1=Help F3=Exit F14=Curr flt F23=	F4=Prompt F5=Refresh =More options F24=Mes	F6=Create F9=Cmd sages	H F12=0	Cancel	

When <Enter> is pressed, the Work with Release Distribution panel is presented:

THNSPT	Change Management System	
	Work with Release Distribution	
Release . : 00097 Target Environment: *L	Acme Pricing Modifications IV Send Type COMS (Coms/Tape/Intl)	
Application . 10	I I ACM ACINE Sales Older Flocessing	
Type 'X' to select dist Target Systems Validation System Training System Live System (*	tribution and type any DMS submit RCVRLS override: Dst Ovr Target Systems X Y FWD *) _	s. Dst Ovr
		Bottom
F3=Exit F4=Prompt 1 F15=Select/Omit CRs 1	F7=Previous APP F8=Next APP F9=Cmd F12=Canca F21=Transfer F22=Packaging F24=Msgs	el

This time, all authorised production systems other than the local production system are listed. Those distributed to from the local, intermediate system, are automatically selected. Those that are distributed to via another route are deselected and marked with the (* FWD *) notation.

9.3.3 Forwarding a Release from the command line

A new command has been introduced to allow releases forwarding to be user initiated. The command is Start Release Forward Transfer (STRFWDTFR) and when prompted appears as follows:

Start Release Fo	orward Transfer	(STRFWDTFR)
Type choices, press Enter.		
Development Centre System: Release Number:		Character value Character value
		Bottom
F3=Exit F4=Prompt F5=Refresh F13=How to use this display	F10=Additiona F24=More keys	l parameters F12=Cancel

When used it is necessary to specify the originating development centre system along with the release number that is to be forwarded.

There is an additional Forward/Auto parameter. This should always be left at its default setting of *FWD. The *AFW (Auto Forward) value is for internal SEE/Change use only and should not be used.

9.3.4 Restrictions

The following points are noteworthy when configuring and using release forwarding.

9.3.4.1 Distribution method

Release forwarding is only supported when the software release transfer request is set to *COMMON. This is the default setting. The software release transfer request is set by general parameter @OMR and the distribution method inspected via the Work with Transfer Requests (WRKTFRRQS) command.

9.3.4.2 Multi application releases

Previously, SEE/Change has allowed the allocation of CRs from different applications to a single release. For applications that do not use release forwarding, this is unchanged. However, once a CR that belongs to an application that uses release forwarding is assigned to a release, subsequent CRs, belonging to other applications, may only be assigned to the same release if the distribution configuration is identical to that of the first CR / application.

9.3.4.3 Target Environment

From the development centre system, the release may be targeted to either the Acceptance Test environment (*ACP) or the Live / Production environment (*LIV) on the target system(s).

However, releases that are forwarded, manually or automatically, from intermediate systems, may only be targeted at the Live / Production environment. In other words, Acceptance Test systems configured on production systems are only supported in the first tier of systems distributed to from development centre systems.

9.3.4.4 Transfer Status Displays

The Work with Transfer Status displays (WRKTFRSTS) will show releases being sent from intermediate systems, via the outgoing view, cycle through the normal communications manager processes (*ACTSND, *RSPW and so on).

9.3.4.5 Confirm Wait

When a production system that is also configured as an intermediate system is set to have incoming software releases confirmed (*CNFW status), the release will not be available for manual forwarding until it is confirmed <u>and</u> installed locally.

However, if the intermediate system is set to auto-forward the release, the onward transmission of the release packet will take place without requiring local confirmation of the incoming release.

9.3.4.6 Remote CR Status Messages

Messages from remote production systems are sent back to the development centre system, regardless of where a given production system received its release packet from.

9.3.4.7 Release Packet Archive

Incoming release packets are always stored locally when the @RPA parameter is specified. For end-of-line systems it is recommended that this parameter is not set.

9.3.4.8 Multiple Logical Intermediate Systems on single Machine / Partition

When an intermediate system forwards a release, the DMS jobs need to restore the release packet, adjust control file values and resave it before forwarding it to the next production system. Because each production system expects the *saved* library name within the received release packet (save file) to be that of the @OMR general parameter (usually O#TFR), this is the library name used for the restore and subsequent save. Consequently this library name will exist on the system for a short time whilst the forwarding transfer request is in the active send (*ACTSND) status.

As a consequence of the above, a conflict will occur if more than one environment within a single machine / partition attempts to simultaneously build and / or forward a release. Only one of the conflicting processes will succeed.

SEE/Change – Supplementary Manual 4.5003

If this occurs, the failing transfer request(s) will gain an *ERRORS status. This will need to be reset and the release forwarded manually from the Release Manager once the successful process has completed and the library name is free to be restored once again.

10 Audit Manager

10.1 Transfer history (4.4002)

A new log has been created to record the instances of an object being transferred from one CR to another. The information is available via the *Display CR* panel with option 25=*Tfr History*. The following picture shows the new Transfer history window.

:	Object Transfer History	:
CR : SY1 000052 / Appl : AP1 Demo appl : Status: *DEV 14/05/98 : 5=Display 11=Tr : 25=Tfr History Opt Object Ref Id : 25 MKRPG RPG : CMAPGM RPGLE	From IR/CR To IR/CR Date Time User 000052 02 000052 03 15/05/98 8:53:02 MARTI	: N : : : : : : : :
	E F1=Help F12=Previous	: ottom : : :
F2-Frit F5-Pofroch F	A=Cmd F11=Change view F12=Cancel F18=Documents	Bottom

It gives basic details of the *from* and *to* CR for the transfer, date,time information and the user that performed the transfer. The list will display all the transfers ever logged for the object in reverse date order.

11 Communications Manager

11.1 Pack size limit (4.3003)

The communications packet size limit field allowed a maximum of 99,999 K. This is too small by today's standards so instead of increasing the field size we have changed the figure to be in megabytes rather the K's. Therefore the new maximum is 99,999 Mb's.

Note: the installation process for this PE will preserve your maximum packet size by converting it to the equivalent size in megabytes.

12 Interfaces

12.1 Hawkeye interface (4.3006)

New commands have been created to facilitate an interface to HAWKEYE the analysis tool. Some implementation work needs to be carried out in that user defined options need to be created in SCDM to access Hawkeye, and user defined options need to be created in Hawkeye to initiate SEE/Change functions. Separate documentation is available with instructions on how to implement this feature.

12.2 Import register upgrade for CASE tools (4.3006)

For CASE tool users the *Update import register* command has been improved upon by the provision of a 'work with' style panel. From this panel you can action all the same options you had to do one at a time with the old UPDIMPREG command. An example of the panel follows:

	THE	NON Change M Maintain Im	anagement system. CR port Register	: 210032 / 10
Twpe options	Pross Entor			
1-Not Poadu	2-Poady	1-Pogistoro	d 6-Imported 7-Deleter	4
I-NOU Ready	2-iteauy	4-Negiscere	a o-imported /-Dereted	٠ ـ
Opt Name	Type	Attribute	Text	Status
-				
0#705	*PGM	RPGLE	Re-bind module to programs	Ready
0#705	*PGM	RPGLE SRC	Re-bind module to programs	Ready
O#705FM	*FILE	DSPF	Select Programs for Module	Not Ready
0#708	*PGM	RPGLE	Bind modules to pgm's	Ready
0#708	*PGM	RPGLE SRC	Bind modules to pgm's	Ready
0#709	*MODULE	RPGLE MOD	Populate module pool entries	s Not Ready
0#709	*PGM	RPGLE	Populate module pool entries	s Ready
0#709	*PGM	RPGLE SRC	Populate module pool entries	s Ready
O#710	*PGM	RPGLE	RMVILEOBJ subfile	Ready
O#710	*PGM	RPGLE SRC	RMVILEOBJ subfile	Ready
O#710FM	*FILE	DSPF	RMVILEOBJ Confirmation Subf	Not Ready
0#711	*PGM	RPGLE	TFRILEOBJ subfile	Ready
				More
Enter F3=Exi	t F4=Prompt	F9=Command	Line F12=Cancel F13=Repeat	
ENCOL TO ENT	e i i i empe	15 continuina	nepeat	

The panel gives a simple list of the contents of the import register for a given CR. The status of any of the records can be changed via the appropriate option. The list can be subsetted by Name, Type or Attribute and function key F13=Repeat can be used to select multiple lines.

This panel accessed via a new menu that can be shown on return from a CASE tool to SEE/Change.

The current method of using UPDIMPREG as a command against objects is still supported therefore any custom code written will still operate normally.

See new parameter @CAS in *General Parameter Changes* for further information.

12.3 JD Edwards World software Interface (4.4003)

12.3.1 Overview

SEE/Change now includes an interface to the JD Edwards *World* software package. JDE programs and files as well as JDE internal 'flexible' data can now be managed through SEE/Change. This means that when you retrieve a program or display/print file you can also retrieve any associated User Defined Codes, Vocabulary Overrides, Processing Options etc. and promote these through the change cycle like any other object in SEE/Change.

SEE/Change now supports seven new object types for JDE-specific items:

Object Type	Description
JDEDD	Data Dictionary
JDEDW	Dream Writer
JDEMNU	Menu
JDEPO	Processing Options
JDEUDC	User Defined Codes
JDESV	Software Versions
JDEVO	Vocabulary Overrides

These object types can be retrieved into a CR and promoted/reverted just like any other object types in SEE/Change even though they do not represent real AS/400 objects but rather records in JDE internal 'flexible' files. For development purposes SEE/Change stores its own copy of these flexible data files in the common Application Message File library. While working on a JDE object type in a CR, developers will be modifying the data in the flexible files in the message file library, and when a CR is promoted, flexible data will be copied (via JDE functions) from this library into the target JDE application libraries. The new object types can be edited directly from within SEE/Change using standard JDE SVR functions.

Current flexible data items (of the new object types) will not be registered in SEE/Change to start with (there is no take-on job equivalent to UPDSRCREG (Update source register)). They will only be registered once they have been promoted to live. After this, they will be registered as non-source based objects and can be prompted for on retrieval etc.

For JDE Applications, SEE/Change will now compile standard AS/400 objects (i.e. PF, LF, DSPF etc) using JD Edwards compile commands. An option to use the JD Edwards SVR/Design Aid for these object types is also supported.

12.3.2 Configuring SEE/Change for JD Edwards

The JDE interface is enabled on a per-Application basis. In the same way that you must create target database libraries for each Application, for a JDE Application you must create a target JDE flexible data file library for each environment (Live, Acceptance, Integration) that you wish to configure. These JDE libraries must contain the 'flexible' files (typically the libraries will be copies of the JDE Common library).

- The Work with Application Configuration dialogue has now changed -
- The Work with Application Details extended options screen (F18), displays two new fields:

JDE Application?

Set to 'Y' to use the JDE interface for this Application.

SEE/Change Version for JDE

Set to '01' for the current implementation of the interface.



After selecting your systems/sites, the **Work with Application Configuration – Libraries** screen will display a new line for each site to allow you to configure the target JDE libraries that you previously set up.

Pressing Enter on this screen will bring up a pop-up window JDE Flexible Files have not been configured. Configure them now? If you reply 'Y', SEE/Change will run the command CPYJDEF which will copy the required flexible files from the Live JDE target library of the Development System/Site into the common Application Message File library. If not all of your flexible files exist in the Live JDE target library or if you wish to configure the files later you can run CPYJDEF from the command line. The next configuration issue is to determine your archive strategy for JDE flexible file data. By default no JDE flexible data will be archived. Archiving can be enabled on a per-JDE object type basis. On the first screen in **Work with Application Configuration**, type option 17=JDE *Info* against the configured JDE application. A list of JDE object types is displayed along with whether or not they are to be archived. Type a 1=Se|ect against an object type to configure it to be archived, or type a 4=De-se|ect to configure it not to be archived.

SY1 Test D JDE Object	evelopment System s to be Archived
Application : AP1	Local Sys. : SY1
1=Select 4=Deselect	
Opt Object Type Archive? JDEDD N JDEDW N JDEMNU N JDEPO N JDESV N JDEUDC N JDEVO N	
	Bottom
F1=Help F3=Exit F9=Cmd F12=Cancel	

12.3.3 Using the SEE/Change - JDE Interface

Standard AS/400 object types (i.e. PF, LF, DSPF, PRTF, RPG, CLP) can be retrieved, viewed, edited, etc. in a JDE Application in the same way as they would be in any other application. Additionally, these object types can be worked on using the appropriate SVR function or JDE Design Aid by typing option 10=Edit JDE against the object.

Standard AS/400 objects are compiled (option 14 or 15) using JDE compile functions. The calls to these functions are stored as execution messages in a new SEE/Change message file OMSMSGJ. The compile commands in these messages will override all other compile commands including user-defined ones in OMSMSGU. To change the default parameters for compiles, edit the execution messages in OMSMSGJ (the messages reference commands that can be prompted for parameter names etc). The pre-compile commands defined through JD Edwards (in file F98CRTCMD) are supported.

The new JDE flexible data object types can be retrieved into a CR in the same way as any other object type. In most cases the name of the object will be self-evident. However, User Defined Codes have a two-part identifier - System code (4 CHAR) + User Code (2 CHAR). To support this in SEE/Change, the two codes have been concatenated to form the first 6 characters of the 10 character object name (the remaining 4 characters must be Blank). The 4 + 2 characters must be entered exactly as they appear on JD Edwards screens, in many cases this will result in embedded blanks appearing in the object name (e.g. the User Defined Code with System = '06 ' and User Code ='BD' must be entered in SEE/Change as '06bbBDbbbb' where b=blank).

Once retrieved into a CR, the new object types can be edited using the appropriate SVR function by typing 2=Edit against the object. Editing one of these objects will result in changes to the data in the flexible files in the Application Message File library. You must ensure that this library is higher in your library list than any other JDE libraries.

(Note that the JDE flexible data in the Application Message File library will be common to all developers, so that if, for example, a change is made to a JDE User Defined Code, this change would be visible to another developer concurrently developing the same Code).

12.3.4 JDE Library Overrides

It is assumed that most flexible files will exist in one library at each level (Live/Acceptance/Integration). However, See/Change will allow the override of these libraries for particular JDE object types. The override libraries can be configured from the **Work with Application Configuration – Libraries** screen. Enter option 12=Obj type overrides against the subfile line for JDE libraries. **The Work with Application Overrides – by Object Type** screen will be displayed and you can enter the library(s) to override to for any JDE object type in any environment. (See also **Enhancements to override processing** above.)

12.3.5 Data Area VTTL@

When calling some JDE functions from within SEE/Change, a check is made on a JDE data area called VTTL@ in QTEMP. When SEE/Change starts up, a call is made to a new exit program O#SEXT. O#SEXT will create and initialise this data area if it does not already exist. If this conflicts with the way your JDE environment operates, you will have to remove/replace this exit program.

12.4 The new ASSET Interface (4.4200)

12.4.1 Introduction

As an integral part of its functionality, SEE/Change has continued to offer interfaces to the most widely used CASE tools. An interface to the ASSET CASE tool has been available since the very earliest releases of SEE/Change. The new ASSET interface builds on the existing functionality and introduces the ability to manage ASSET definitions within SEE/Change. This new interface was designed with assistance from SSA and one of the largest BPCS users.

By applying the new interface, AS/SET definitions may be checked out from a live set to a development set under the control of SEE/Change. In the development set, programmers may change definitions and create the related AS/400 objects. Promotion under the new interface causes both the AS/SET definitions to be promoted to the target testing set and the AS/400 objects to the associated target environment libraries. The new interface extends all the standard archiving facilities to ASSET definitions allowing older versions to be saved and even reverted. Standard controls apply to ensure that a promotion to live signals the end of the development cycle and causes all locks on the ASSET definitions to be removed.

12.4.2 System Configuration

For every application environment to be configured, a corresponding ASSET Set is expected. Use ASSET function *Edit Application Set* to include SEE/Change system libraries OMSOBJ and OMSDTA in the set library list of all sets to be used in SEE/Change applications.

Ensure the ASSET subsystem job description (ASSETO/ASSETJOBD) includes SEE/Change libraries OMSOBJ and OMSDTA. If the ASSET subsystem is active, terminate and re-start the subsystem.

To inform ASSET that the interface is active, change ASSET data area as follows:

```
CHGDTAARA DTAARA (ASSETO/ASSETDTA (256 1)) VALUE ('Y')
```

Ensure that the correct ASSET save library is specified in position 148 of DTAARA ASSETO/ASSETDTA. The interface programs derive the name of this library from the data area and make reference to it. If the ASSET save library has been renamed, ensure this data area holds the correct name of an existing ASSET save library.

Ensure that the libraries OMSOBJ and OMSDTA are also reflected in the data area OMSDTA/ADKLIBL. Please ensure that a space is left between each library name specified on the data area. The starting position of each entry is unimportant.

Change the base SEE/Change job description OMSDTA/OMSJOBD to add the ASSET libraries to it. Clearly, if the names of these libraries have been changed, these new names must be reflected in the library list. After these changes have been made, the INLLIBL list of OMSJOBD should look similar to that shown below:

QTEMP OMSDTA OMSOBJ OMSSAV ASSETO ASSETF QPDA QGPL

If you wish to ensure that objects imported from ASSET are not edited within SEE/Change, ensure you set the parameter @AIE appropriately. To prevent editing of ASSET sources type the command WRKPRMDTA and set @AIE to *NO.

Before you continue with any subsequent steps, exit SEE/Change and then re-start it to ensure the changes made have taken effect.

12.4.3 Application configuration

The configuration of an ASSET application in SEE/Change is not dissimilar from that of a standard application. Some additional steps are required and these are described below. It is assumed that users will be familiar with the process of creating an application. Standard application setup is described in the SEE/Change manual p. 2-59 but, if any advice or clarification is needed, please contact Thenon Support.

From the *Work with Application Configuration* screen, (WRKAPPCFG), take the new option *6=ASSET Cfg* against your ASSET application in order to specify the correct sets for promotion and retrieval:

ADKCFG	D		Work with	AS/SET configuration	QPADEV0001 THENON	12:01:45 28/02/01
Applic 1=Cre	ation Sys ate 2=Upo	stem . date 4	: SY1 =Delete 5=Disp	lay 8=Position		
Opt	Sequence	Mvmt. Type	Set	Object Library		
-	1 2 3 4 5	*DEV *MDL *ACP *LIV *LCK	TESTDV TESTTS TESTAC TESTLV TESTBS	TESTDVO TESTTSO TESTACO TESTLVO TESTBSO		
F3=Exi	t F7=Pag	ge up	F8=Page down	F12=Cancel		

Sequence:

This is the search sequence that the retrieve program will look into, when looking for an AS/SET definition.

Movement type: *DEV corresponds to the set where development takes place.

*MDL is the set where definitions reside when promoted to Integration.

*ACP is the set where definitions reside when promoted to Acceptance.

*LIV is the set where definitions reside when promoted to Live.

*LCK is the set where the original definitions are held.

Set: This is the set in AS/SET where definitions are held. Named sets must already exist. Sets to which definitions will be promoted must not reference any other sets.

Verify that your application is correctly configured as an ASSET application, and that the correct development set name is specified, as in the example below, by taking option *5=Display* from the *Work* with Application Configuration screen, (WRKAPPCFG), against your ASSET application:

THN170 SEE/Change Version 4.4200
Work with Application Details
Application code : SAL
Application description BPCS V6.04
Development centre system(P). DEV Dev System
Development centre site(P). DEV Prd Env
Application ASP $\dots \dots \dots$
Application job description SALJOBD
Application message file library . OMSSAVSAL
Default CASE tool (P). *ASSET Model/Set Name TESTDV
Cross-application documentation . N (N) one/(A) ppl/(D) ate
Re-compile objs at remote sites? . N (Y)es/(N)o
Re-compile objs at Dev Centre? N (Y)es/(N)o
Distribute source code? N (Y)es/(N)o
Distribute object override info? . Y (Y)es/(N)o
Message file operations D (M)erge/(D)uplicate
Multiple versioning? Y (Y)es/(N)o
Planned concurrent development? . Y (Y)es/(N)o
Auto revert if promote errors? N (Y)es/(N)o/(P)rompt
Number of archiving levels 01 (0-99) Archive Library
Configuration complexity 3 1=Simple 2=Intermed 3=Complex More
F1=Help F3=Exit F4=Prompt F9=Cmd F12=Cancel F16=Update F24=Messages

Ensure that source files and environment libraries have been correctly configured for non-ASSET object types.

Ensure that every application job description specifies the ASSET product libraries, (ASSETO and ASSETF), on its initial library list, (INLLIBL).
Define the application source files in the usual way and ensure a source file is declared for ASSET batch, display and file definitions, types ADKBAT, ADKDSP and ADKFIL. The default source file names use the prefix 'Q..', i.e. QADKBAT, but these may changed if alternative naming conventions are required. (Do not use existing source files QCLSRC, QDDSSRC etc. ASSET definitions must be held in their own dedicated source files).

Existing ASSET definitions in the configured Live set may now be registered into SEE/Change by calling the respective commands, BATMBR for batch definitions, DSPMBR for display program definitions and TPFMBR for file definitions. Each program requires three parameters, the Live set name and associated object library and the source file name. Before calling these programs ensure that each source file defined in your application configuration for your ASSET object types contains a single member named DEFAULT which holds one blank line. After these take-on programs have completed, the source update should be run, (UPDSRCREG), to complete the registration of the AS/SET definitions into the application. Please be aware that if you need to re-run any of the take-on programs it will be necessary to first clear the source file specified on the command.

From the Configuration Manager menu, select option 22=User Enrolment, (WRKUSRAUT). Ensure that all ASSET developers are authorised to the function WRKIMPREG.

From the Configuration Manager menu select option *11=Work with Parameter Data*. Ensure the parameter @CAS is set as in the example below. The 'KEYWORD' specified should be the three letter code of your ASSET application, and the 'DESCRIPTION' should be set to 'Y'.

SI	EE/Change Version 4.4200	
D	Work with Parameter Data	
Attr: *LEN: 3 / 1 A *DSP: *STD *EDT: *STD	*VLD: *STD *VL1: *VL2:	
Enter/update parameter tak	ble: @CAS : Case Tools Import Prompt Flag	Y/N
Parameter Var I KEYWORD I	Parameter Value DESCRIPTION	
SY1 Y	Y	
		Morro
		More
F1=Help F3=Exit F9=Cmd H	F12=Cancel	

Create an application *AFTER program in a CR under your new ASSET application. In the source member remove the comments from the source lines which relate to the ASSET interface. These are clearly marked. Promote the *AFTER program to the Live Development library. This *AFTER program is an important part of the interface since it provides the hook into the ASSET functionality. For this reason, it is crucial that this program is put to Development Live before any ASSET definitions are retrieved and moved.

At each Production system where an ASSET application has been defined, the 'pseudo' SEE/Change ASSET definitions as they are defined in SEE/Change will be delivered as source members to files named ADKBAT for batch definitions, ADKDSP for display program definitions and ADKFIL for file definitions. Users may wish to configure application level, object type overrides for each of these related ASSET object types so that these source files and members are not created into the Live Production library. No other action is required at a Production system.

Your ASSET application is now ready for use.

12.4.4 Recommendations for Best Practice

Ensure that any of the sets defined that are not development sets are properly secured with the correct OS/400 authorities. If users need to look at a definition they **must** be required to bring it into the Development set using SEE/Change only.

The Transfer facility must not be used in any ASSET CRs. It is possible to prevent users from having any access to the transfer facility by disabling the use of function TFRCROBJ against their profiles, (WRKUSRAUT). Bear in mind that revoking this function will also prevent a user from transferring objects from a non-ASSET CR.

12.4.5 ASSET object types

Here is a list of the ASSET object types:

- ADKBAT: Batch program definitions. These objects may be imported into SEE/Change and promoted through the SEE/Change development lifecycle.
- ADKDSP: Display program definitions. These objects may be imported into SEE/Change and promoted through the SEE/Change development lifecycle.
- ADKFIL: File definitions. These objects may be imported into SEE/Change and promoted through the SEE/Change development lifecycle.
- ADKRPT: Report program definitions. These objects may be imported into SEE/Change and promoted through the SEE/Change development lifecycle.

- ADKMDL: Data model definitions. These are required when creating a new ASSET programs and when defining new files. Model definitions need to reside only in the Development set and are not required to be promoted through the interface. Consequently, promotion of Data model definitions is not supported by the interface.
- ADKSUB: Repository subroutine definitions. When base BPCS is currently delivered no objects of type ADKSUB are supplied. This makes it unnecessary to promote subroutine definitions through the interface. Consequently, promotion of Repository subroutine definitions is not supported by the interface.
- ADKFLD: Field definitions. It is expected that field definitions will not be regularly changed. Due to the large number of dependencies, it is expected that developers will be required to create new fields rather than editing existing ones. For this reason, field definitions only need to reside in the Development set and are not required to be promoted through the interface. Consequently, promotion of Field definitions is not supported by the interface.

12.4.6 Using the Interface

The object level interface between SEE/Change and ASSET is implemented through the import facility. ASSET definitions are now registered and moved by the new interface. Special asset object types have been created in SEE/Change to represent each type of ASSET definition. These are, in fact, simply source members but it is the movement of these pseudo ASSET object types in SEE/Change that triggers the movement of the true ASSET definitions:

ASSET session is invoked through *Change Manager* function WRKCHGRQS (Work with Change Requests).

When you create a new CR via Change Manager function WRKCHGRQS, you are prompted to specify field: CR CASE Tool. The default is set to the value you have specified for Default CASE tool in the application configuration. You can prompt for other valid values using F4. The value of *ASSET indicates that development is ASSET based. The value of *NONE indicates that the development is RPG based.

Before working with ASSET, first retrieve or create all the ASSET definitions required in the ASSET CR. New object types are provided in SEE/Change with the prefix 'ADK..'. For information on the new ASSET object types supported in SEE/Change, see the section below entitles 'ASSET Object Types'. Please be aware that the retrieval of an ASSET definition instigates background processing within ASSET and will take longer than expected for the more familiar standard source retrieval.

Option 13=Wrk CASE (or function WRKCASTLS) provides the entry point into an ASSET development session. Command INTERAST is prompted for you to select the AUTHOR parameter. The set name is supplied by SEE/Change based on the application configuration.

Batch submission must be used for all object generation within ASSET. This is necessary to ensure that records are correctly written to the Import Register which identifies the generated objects to SEE/Change.

The generation of ASSET logical files required that the based-on physical file is first brought down into the development set. Clearly this is a requirement of a change managed structure where definitions are moved from one set to another in the course of the development lifecycle.

ASSET files must be imported into SEE/Change before generating related programs. Only after these files have been imported into SEE/Change should ASSET programs be generated and imported.

When objects are regenerated, their details are recorded in the import file.

When control is returned to SEE/Change, an exit menu offers two functions: 1. Work with Import Register and 2. Perform full Import. Is is recommended that the Import Register be examined before importing into SEE/Change. On generation of ASSET objects many additional objects are generated and working with the import register provides the opportunity to remove these. A new option, 8=Remove, is now offered on the 'Maintain Import Register' screen to remove unwanted objects.

THENON Change Management system. CR : 000016 / 01 Maintain Import Register					
Type options, 1=Not Ready	Press Enter 2=Ready	4=Registere	d 6=Imported	7=Deleted	8=Remove
Opt Name	Туре	Attribute	Text		Status
ASBAT1 ASDSP1 ASDSP1F CLMSG OBJ069B OBJ069BC OBJ069BD OBJ070B OBJ070BC OBJ071B OBJ071BC OBJ071BD	*PGM *PGM *FILE *PGM *PGM *FILE *PGM *PGM *PGM *FILE	RPG RPG DSPF NOSRC RPG CLP DSPF RPG CLP RPG CLP DSPF	ASBAT1-Batch Pro ASBAT1-Display B ASDSP1F CLMSG OBJ069B AS/400 S OBJ069B AS/400 OBJ069DB AS/400 OBJ070B AS/400 S OBJ070BC AS/400 OBJ071BC AS/400 OBJ071BC AS/400	ogram 1 AS Program 1 AS Software Eng Software Eng Software Eng Software Eng Software Eng Software Eng	Ready Ready Imported Imported Imported Imported Imported Imported Imported Imported Imported More
Enter F3=Exi	t F4=Prompt	F9=Command	Line F12=Cancel	F13=Repeat	

Having made changes to the Import Register, all required ASSET objects may be imported into SEE/Change by performing a full import. (Option 2 on the exit menu).

Because SEE/Change is informed of the generated objects only after their construction, (as opposed to the standard object retrieval mechanism), CR locks can only be determined when attempting to import these objects. If an object of the same name is already registered under a different CR, a locking problem will occur, and the import run will end unsuccessfully. If an import file entry cannot be processed due to the objects being already locked to another CR, you should remove it from that other CR, and then re-execute the import function.

Be aware that the CR cannot be moved to the testing or live environments unless all entries for that CR on the import register are fully imported.

If it is necessary to delete any ASSET definitions in a CR, it is required that the ASSET definition is deleted before the associated file or program object can be deleted.

After all objects have been successfully imported, change management cycle is as per SEE/Change standards.

Archiving continues to exist, where configured, for all ASSET CR's. The new ASSET interface ensures that true ASSET definitions are also included in the archiving mechanism allowing old definitions to be archived and reverted back to the live set through the normal promotion and reversion of a CR.

12.5 COOL:2E - V7.0 ILE compatibility (4.4201)

Version 7.0 of COOL:2E is able to call the RPGIV compiler and generate ILE objects. With PE4.4201, the SEE/Change *SYNON interface can now import and promote the generated ILE objects and their sources.

Note: The COOL:2E RPGIV ILE generator can be toggled to generate either modules or programs. For programs, the command CRTBNDRPG command is used; for modules, CRTRPGMOD is used. If you have configured your SEE/Change application to re-compile objects, use the *Edit Compilation Commands* and the *Edit ILE Module/Program Commands* functions in the *Configuration Manager* to ensure that for RPGLE_SRC, the CRTBNDRPG and CRTRPGMOD commands correspond with the COOL:2E *CRTBNDRPG and *CRTRPGMOD messages in the *Messages file (Y2U1022 and Y2U1024 respectively).

12.6 LANSA variables (4.4004)

SEE/Change has an object name length of ten characters. This prevented LANSA users from creating variable names of greater length. SEE/Change has been changed to allow the successful import and promotion of long variable names. It achieves this by storing the long name in the text description of the SEE/Change registry entry. The first 10 characters of which are still used as the SEE/Change object name. Since SEE/Change is still using the object name the first ten characters must still be unique but SEE/Change converses with LANSA import / export routines using the extended name stored in the objects description.

12.7 High Availability Compatbility / EXCCRLST (4.4301) (4.4301)

SEE/Change is often used in conjunction with high availability products hosting a variety of configurations. However, SEE/Change does not have specific programmable options for the handling of file movements in the High Availability environment. Instead, native SEE/Change behaves as follows:

- Create temporary library [archive]
- Move old version of file into temporary library
- Deliver new object to target library
- Copy data from file in temporary library to file in target library
- [4.4300 and earlier] Apply journal to file in target library according to version in temporary library
- [4.4301 and later] Apply journal to file in target library according to version in temporary library <u>unless</u> a journal is already present on the new file. The assumption is that this can only be due to a high availability product.
- Remove journal from file in temporary library [4.4202]

In CR reversion complementary processing is carried out

- Create temporary library
- Move version of file from target library into temporary library
- Move previous version of file from archive library to target library
- Delete archive library
- Copy data from file in temporary library to file in target library
- Apply journal to file in target library according to version in temporary library
- Delete temporary library

Sometimes the above behaviour can conflict with High Availability products and for this reason it has always been recommended that high availability be suspended for database object for the duration of a promotion cycle. It will now be possible to automate this process, subject to the high availability product being command driven, by implementing the Execute CR Objects List (EXCCRLST) command into *BEFORE and *AFTER processing programs.

This new command allows the user to select a subset of object types and execute a user defined command for each. In the high availability environment this can therefore be utilised to suspend and re-establish high availability for a planned file movement.

The command syntax is shown below:

Execute	CR List (EXC	CRLST)
Type choices, press Enter.		
Development Centre System IR Number	*LIBL *ALL *ALL *ALL *ALL *ALL *BOTH *ALL	Character value 000001-999999 01-99 Name, *LIBL Name, *ALL, generic* Character value, *ALL Character value, *ALL *ALL, *DB, *PGM, *JDE *BOTH, *YES, *NO *ALL, *COMPILE, *CPYREF
F3=Exit F4=Prompt F5=Refresh F24=More keys	F12=Cancel	Bottom F13=How to use this display

The last parameter accepts a command that will be run for each of the object selected by the other parameters. Substitution values are supported as follows:

- &1 Object Name
- &2 Object Library
- &3 Object Type
- &4 Object Attribute

13 Supplemental PE Note Information

This section details supplemental PE note information.

13.1 General Parameter Changes

The following table describes parameters that have been added to SEE/Change or changed by release.

Parameter	In Version	Description	Values	Default
@CDR	4.3003	Allows a CR to be promoted directly to status 'ready' without having to promote to 'test' first.	*YES *NO	*NO
@NDO	4.3004	Specifies the owning profile of CR libraries not in development.	*SAME USRPRF	*SAME
@DCH	4.3006	Specifies the prefix character on source documentation. i.e. previously hard coded as V eg V001. Can be specified at application level.	Any single Character.	V
@CLT	4.3006	Declares what type of library a CR library is created as.	*PROD *TEST	*PROD
@SNT	4.3006	Defines default Send Type in Work with Releases.	COMS TAPE	COMS
@CAS	4.3006	Determines whether CASE tool options menu should appear on return from CASE tools SYNON and AS/SET. The options menu allows you to run the new <i>function Work with import register</i> and or perform an IMPORT *FULL	By app code then "Y" to activate menu.	
MONR	4.4000	Specifies the number of seconds QMONR waits between looking for incoming releases.	Number of Seconds	300
@AAR	4.4001	Allow acceptance retest.	YES	
		Prevents a CR being reverted from live if it is being redelivered to acceptance on a production site. Thus the same CR resides in both acceptance and live.	NO	
		Do not use this option without speaking to Thenon support.		
@LXA	4.4002	LANSA XREF Active. Allows people who use LANSA to develop database objects but maintain program	YES NO	

		objects conventionally to utilise cross referencing and perform automatic compiles based on LANSA file movements. This requires a small amount of implementation work, for which please read the help text for the parameter for further details.		
@AUD	4.4200	Authorise Distribution to Systems? Set this parameter to '*YES' if you want to specify which users can send Releases to a	*YES *NO	
		function from within Work with Systems Configuration.		
@IFP	4.4200	IFS work directory prefix.	Any	SEE
		This parameter is analogous to @WRP (Work Library Prefix). If @IFP was set to 'SEE' and your development system's site code were 'SY1', all locally developed applications would utilise the root IFS directory <i>/see#sy1</i> on that system. If an application developed on system 'SY2' sends IFS files to 'SY1' they would be stored in the directory <i>/see#sy2</i> .		
@CAU	4.4200	*PUBLIC Authority to CR library.	*USE	
		Authority given to *PUBLIC by APYCRAUT. (Note: *SYSVAL will retrieve QCRTAUT authority)	*EXCLUDE *ALL *SYSVAL	
@SNL	4.4200	Send to Live from <i>Work with Change Requests</i> . Set this parameter to *YES if you want to be able to send a CR to a Live production system from the <i>Work with Change Requests</i> function. A new option 48=Send to Live will be enabled, and typing this option against a CR will cause a new Release to be automatically created and the CR allocated to it. You will then be prompted for the system(s) to which the Release is to be sent. (Note: the CR will also be promoted to Live on the Development machine.)	*YES *NO	
@ROV	4.4201	Allow overrides for DMS submit RCVRLS? Set this to *YES if you wish to override the default job description (and/or schedule dates/times) used by the DMS job when it submits a Receive Release. Until the parameter is changed, a blank entry is equivalent to a value of *NO.	*YES *NO	
@AIE	4.4004	ALLOW IMPORTED EDIT For AS/SET users. This flag allows you prevent the editing of generated source that has been imported into the CR. When the user attempts to edit the source via SEU an error message will result.	*YES *NO	

@NDO	4.4004	NON DEVELOPMENT OWNER.		
		This parameter no longer applies to the MDL/Integration environment if you utilise <i>the Updating MDL/Integration with changed objects only</i> as described earlier in this document.		
@SDL	4.4301	Suffix for restore of DMS libraries	01 to 99	
		This parameter defines a suffix that will be added to the library name that is calculated by the comms manager when restoring release packet libraries. This is useful when the same release packet is received by multiple production partitions (instances of OMSDTA) that reside on the same machine.		
@ARR	4.4301	Allow reset of Ready for Release	YES	NO
		This parameter, if set to YES, allows CR's at Ready for Release status to be reverted to their previous environment (Module/Integration or Acceptance) without having to perform a redevelopment movement first.	NO	
@USD	4.4304	Update Object's Source File Details	YES	NO
		When set to YES, SEE/Change will update the OS/400 source file attributes of simple compiled objects with the source file and library into which the source is filed.	NO	
@RPA	4.4305	Release Packet Archive Library	Library name	
		Specified the library name in which release packets are stored by production systems that are also intermediate systems.		
@RFC	4.4305	Release Forwarding Control	NO	
		When set to YES, SEE/Change will check the status of the CR's in the release being forwarded on the development centre as well as the local system. If any are found to be at a status other then Ready for Release or Live, SEE/Change refuses to forward the release.		

13.2 Object type support

The following table describes object types that have been added to the SEE/Change repertoire of supported types by release.

Object Type	Support Introduced in release	Comments
OVL	4.3004	Overlay
PAGSEG	4.3004	Page Segment
PNL-NOSRC	4.3004	Panel Group (No Source)
SBSD	4.3004	Subsystem Description
SRT	4.3004	Program (Source)
TCPIP	4.3004	Program (Source)
DTAQ	4.3006	Data queues are now supported within SEE/Change. They can be retrieved into a CR and promoted in a similar fashion as data areas.
RPGLEREF	4.4002	SEE/Change handles /COPY members for RPGLE in the same way as RPGREF. The object type is RPGLEREF.
TBL	4.4200	SEE/Change handles tables in a similar way to other source-based objects. However, as the source for a table can only contain the table data, source documentation added by SEE/Change is not valid. If you want SEE/Change to handle tables, use WRKPRMDTA to ensure that the parameter @DOC (Document Source) is set to *NONE.
SQL types	4.4300	SQLTABLE, SQLINDEX, SQLVIEW, SQLPROC, SQLTRIGGER, SQLFUNC and SQLDML
DDMF	4.4301	DDM Files

13.3 Software Performance Reports

Software Performance Reports by SPR Number within release:

SPR Log Number	Resolved in Release	Description
355	4.2001	SEU Now available in WRKINVRQS
369	4.2001	LANSA Movement ignoring move to module if no acceptance environment configured.
374	4.2001	Filter validations
378	4.2001	Incorrect text being copied from previous IR in WRKINVRQS
379	4.2001	Program failure if unauthorised to use F6 create in WRKCHGRQS
320	4.3000	Unable to retrieve a source member with a dot in the name
398	4.3000	After program template altered to cater for DLO's
408	4.3000	Ability to create a closed release
412	4.3000	TFRCROBJ failure if target CR already has source member
417	4.3000	CPF4029 and CPF4011 during differing length source movements
418	4.3000	PLI source header size alteration
420	4.3000	Prompt compile fails if run as a batch job
421	4.3000	Originating date validation in WRKCHGRQS
436	4.3000	Resetting of TALS when reverting CR's
437	4.3000	Compile overrides error at 4.2000
451	4.3000	Failure concurrently editing SEU. (Failure prevented, SPR not fully resolved)
453	4.3000	Failure concurrently creating O#SRC1. (Procedure Tightened)
459	4.3000	CR Status resetting for CASE tools (Other than LANSA).
1055	4.3000	Add programmer options (Enhancement)
1070	4.3000	F16 source member search not subsetting type correctly

1185	4.3000	Show programmer name in source comment box (Enhancement)
434	4.3002	XTR File failure during compilation of a physical file.
381	4.3003	Requested Enhancement: Refer to IR
413	4.3003	Requested Enhancement: Ability to set CR to status 'ready' in one step
429	4.3003	Maximum pack size not large enough (Changed to be in Mb's)
439	4.3003	Tape/Coms flag in WRKRLS resetting
461	4.3003	Function key text anomoly in WRKCRDEV
471	4.3003	Able to edit non registered source loop hole in WRKCROBJ
472	4.3003	Multiple module databases causing logical file access path error.
473	4.3003	CHGOBJDFT not working for ILE types
477	4.3003	Session or device error in WRKRLS
480	4.3003	WRKOBJAUT should not allow entry of generic names
482	4.3003	Position to error on empty subfile when creating ILE program
498	4.3003	New program name validation for ILE programs
500	4.3003	N/Fb error when resolving ILE program composition
501	4.3003	Group profile ignored when checking TAL
504	4.3003	Modification ID not being stamped on ILE source
508	4.3003	Requested Enhancement: Show local system on WRKAPPCFG
514	4.3003	BCM movement causing *E71 on RDV from live
528	4.3003	CR's not sequencing properly during release build
530	4.3003	Error occurring during ILE program create with 80+ modules
533	4.3003	BND movement causing *E72 in certain circumstances
403	4.3004	Scan database relations (SCNDBREL) Not recognising SQL programs.
435	4.3004	Illegal answering of TAL authorisation request messages.
531	4.3004	Enhancement: Provide browse facility in object authorities panel.

539	4.3004	Subfile positioning fault in SCDM when ILE is switched off.
540	4.3004	ILE compilation procedure not recognising SQL programs.
545	4.3004	V3R1M0 outfile processing error causing ILE programs to be unseen in SCNDBREL.
554	4.3004	Transferred object from Mdl/Int to Mdl/Int not moved forward to acceptance.
555	4.3004	Enhancement: Need to be able to edit module compilation messages.
560	4.3004	ILE Check CR process registering non ILE programs.
564	4.3004	Not Retaining physical file attributes when compiling.
568	4.3004	Session or device error on ENTER on empty update ILE program subfile.
570	4.3004	Enhancement: Provide CR number as position to field in WRKCHGRQS.
573	4.3004	Configured bar options do not appear on ILE specific WRKCROBJ.
576	4.3004	Module/Integration database' processed alphabetically instead of Development first.
586	4.3004	Compile error when creating ILE program with large number of modules.
587	4.3004	Module pool refresh error when limit of 500 modules reached. (Limit raised to 5000).
592	4.3004	Incorrect text for part in SCDM when overrides exist and marked for delete.
595	4.3004	Prevent two users simultaneously updating IR text in WRKINVRQS.
601	4.3005	Provide better help for message OME3539 for receive release
612	4.3005	Incorrect Database object movement when base module integration lib has been overridden to another lib
613	4.3005	Copy not copying text when IR is already in use by somebody else in Work with Investigation requests
614	4.3005	Prevent CRTPGM compile from happenning when F12 is pressed
615	4.3005	BLDILEREF for site specific objects fixes
616	4.3005	Correct source type & qualifier confusion on some modules in a program that mixes base and site modules
617	4.3005	*E10 occurring during *BCM movements due to internal file size reaching limit.
371	4.3006	Enhancement: Give warning message when editing a MSGF via incorrect function
485	4.3006	Enhancement: Give prompt for full import on return from CASE tool
559	4.3006	Enhancement: Give housekeeping functions for CR and Archive libraries

589	4.3006	Enhancement: Add font to list of preserved PRTF attributes
593	4.3006	Irrelevant member name in programmer options
607	4.3006	Enhancement: Soft code ILE compilation strings
625	4.3006	Enhancement: Allow CRTCRLIB to create library as type *PROD or *TEST
627	4.3006	WRKRLS function keys dissappear when using RUMBA
628	4.3006	Enhancement: Allow user to protect member text in SCDM
629	4.3006	Enhancement: Allow soft coded version number prefix
639	4.3006	WRKINVRQS fails if F6 pressed immediatley after a deletion
643	4.3006	Remove need for library QPDA in CR library list
646	4.3006	See 629
651	4.3006	Delete process executed when should have been a *PRV movement
656	4.3006	Newly created IR text lost if LSTINVRQS is run immediatley after creation
659	4.3006	ILE source documentation overwrites compile time array definition type CTDATA
676	4.3006	Problem When convert OPM RPG to ILE RPG keeping same object name throughout
683	4.3006	Entry module name stored in 9A field instead of 10A
684	4.3006	Reusing CR after a purge causes ILE references to be lost
685	4.3006	As 684 prevent reuse of CR
696	4.3006	Second level text on message input is truncated at 988 characters
700	4.3006	*E03 movement error when no module environment configured
701	4.3006	Conversion places incorrect edit type on source member
708	4.3006	WRKFILTER crashes if called recursively
720	4.3006	Enhancement: Prevent clear of source pools when entering new one
721	4.3006	Enhancement: Source pool locked flag to accept blanks
723	4.3006	Error renaming overlays on *RLS movement
296	4.4000	OCL Movements as detailed earlier in this document

389	4.4000	Do not allow programmer to allocate CCD CR to release if not authorised to WRKRLS
457	4.4000	Enhancement: Allow multiple devices on RCVRLS
562	4.4000	Enhancement: Make QMONR wait period variable
622	4.4000	Authority Catch 22 for TAL's
722	4.4000	Fault in TFRCROBJ when processing DFU's
732	4.4000	Fault when changing entry module of ILE program to one of a different type
736	4.4000	See 622
745	4.4000	UPDSRCREG failing to reorganise XAO if in use
746	4.4000	IMPORTLIB not importing binder directories
776	4.4000	Enhancement: Add release number to retrieve source panel
779	4.4000	EXCMBRLST fails when more than 999 members selected
792	4.4000	SCNDBREL fails when more than 99 Logicals found
797	4.4000	WRKINVRQS display mode not showing text
802	4.4000	Cross referencing anomoly when no Module/Integration environment configured
804	4.4000	BNDDIR incorrectly configured in SEE/Change internal files
811	4.4000	Emergency concurrent development of marked delete objects
827	4.4000	WRKLIBSTS fault when no control record
828	4.4000	Rebuild information in WRKLIBSTS not working See 827
840	4.4000	Enhancement: Function to reset release packet names
841	4.4000	Authorisation anomoly in WRKIMPREG
846	4.4000	Authority incorrect for library operations
847	4.4000	Filter selection window not appearing
448	4.4001	Enhancement: Prevent mandatory back out from production
481	4.4001	WRKCHGRQS filter change not working
840	4.4001	Enhancement: Provide exit program in RCVRLS

858	4.4001	OME1826 occurs browsing relations for site specific object
859	4.4001	Option 26=Object browser not working for objects without an attribute
861	4.4001	Enhancement: Access object browser outside a CR
863	4.4001	CR Extension panels not executing
867	4.4001	DTAARA not being cross referenced correctly
868	4.4001	Embedded SQL objects not being cross referenced correctly
869	4.4001	As 868
872	4.4001	Object browser cannot view bound program source
873	4.4001	As 868
875	4.4001	CPF 5032 occurring during refresh of cross reference data
884	4.4001	CPF5D0D during UPDPGM on ILE delivery to production machine
889	4.4001	Blank export source parameter on ILE program on production machine. OS/400 blanks this parameter out.
899	4.4001	Service program not being cross referenced correctly
902	4.4001	MCH1210, error in warning message display dialogue when more than 999 warnings exist in a CR.
2169	4.4001	As 448
575	4.4002	Enhancement : Object Transfers between CR's not recorded.
605	4.4002	Transferred CR source can no longer be promoted due to source change date anomaly
2527	4.4002	REFXREF failure when more than 9999 object in a library
2528	4.4002	Cross reference exception report printing for every object movement during promotion
2532	4.4002	REFXREF application *ALL not working
2534	4.4002	ILE programs not allowed to promote if another version exists in target library
2540	4.4002	TGTRLS for ILE CRTPGM incorrect for development centre on V4R1M0
2541	4.4002	O#CRI, O#CRLL01 and O#CRFL7 delivery error in 4.4000
2555	4.4002	Missing help text in library status control panel
2581	4.4002	Programs marked for Deletion disappearing from CR when new ILE program of same name is created.

2692	4.4003	Unable to create a user defined reference in cross referencing
2693	4.4003	Objects during movement keeping *PRV status after RDV movement.
2696	4.4003	'Object not valid' message from REFXREF command
2699	4.4003	Recompiles being immediately deleted during RDV
2703	4.4003	Message construct error when restructuring ILE program
2707	4.4003	Locking problem on file XAL caused by multi-threaded promotes
2710	4.4003	REFXREF command with incorrect object type default
2729	4.4003	Fixes to cater for SQLRPGLE_S
2730	4.4003	Superfluous rebinds performed on promote to live of ILE programs
2733	4.4003	Module objects failing to have their movements stamped RDV (See 2693)
2738	4.4003	Prevent 'Marked deletes' removing source from locked source pools
2700	4.4004	Compile directives failure using the DTAMBRS option on files
2713	4.4004	Compile directives in SDA, RLU
2776	4.4004	LANSA parameters (As described in this document)
2855	4.4005	As 2991
2907	4.4005	Cross referencing incorrectly using MBR keyword when precompiling a PRTF
2974	4.4005	Unable to revert a message file due to concurrent development
2990	4.4005	ILE errors at V4R4M0
2991	4.4005	Archive packaging not working on production machine
3040	4.4005	CVTOMSDTA incorrectly looking for library &OMSDTA
3065	4.4005	Overriden object at MDL/INT not being removed on subsequent ACP promote
3097	4.4005	Unable to find service program in library list
3064	4.4101	ILE program assembly error *E72
3199	4.4101	Blank module library causing rebind error (as 3064)
3207	4.4101	Duplicate data error during copy when file overridden to include data.

3234	4.4102	AS/SET import incorrectly registering SQLRPG as RPG
3289	4.4102	Copy of file members always targetted at member *FIRST – changed to *FROMMBR
3292	4.4102	Movement mechanism faling on ILE module rebind if program structure
3294	4.4102	FTP distribution could not process more than 8 simultaneous target systems. Change to maximum of 65.
3313	4.4102	Array index error in function WRKLIBSTS
3407	4.4103	CPF9801 on MOVE *IAC to *LIV
2834	4.4200	OMSMSGU msg ids not recognized
2835	4.4200	Object overrides not maintained
2862	4.4200	XCR record lock causes O#750 to abort
2937	4.4200	*BYPASS not working at PE 4.4003
2952	4.4200	Xref needed for CR Type overrides
2966	4.4200	Show CR Statuses on Work with Releases subfile
2969	4.4200	Filters @ 4.4004 *MDL fails
2977	4.4200	RCVRLS help text not updated
3057	4.4200	Parameter code PSBM
3065	4.4200	Overridden program not being deleted from previous environment
3086	4.4200	MOVCR not reporting error CPD3166
3089	4.4200	New authorisation for users to send Releases
3146	4.4200	APYCRAUT enhanced to cater for QCRTAUT *PUBLIC authority
3205	4.4200	Precompile statements containing '+'
3227	4.4200	3 override libraries necessary for Include Data
3240	4.4200	View DBM commands after DEV
3271	4.4200	Cross-referencing not handling additional sites
3289	4.4200	CRTDUPTQ using TOMBR(*FIRST) on multi-member files
3317	4.4200	E65 on cross referencing re-compile

4.4200	No text when call CRTPGM
4.4200	After converting RPG program to RPGIV, program is not being delivered to Live
4.4200	Transfer goes to *ERRORS (when sending to more than 9 systems)
4.4200	SRVPGM references Live
4.4200	*E70 Invalid library name (CPD0078)
4.4200	Restrict access to SEECFG2?
4.4200	THNNETF send > 9 systems
4.4200	CRTDUPDTQ at 4.4102 – data problem
4.4200	Option 47 not working in SCDM
4.4200	CRTPGM *ENTMODTXT - CPD0047 !
4.4200	XREF incorrect del frm CR @RDY
4.4201	No prompt for CRTxxxMOD cmd
4.4201	Overrides show too many levels
4.4201	Option missing from WRKRLS menu
4.4201	Service pgm – allow UNRSLVREF
4.4201	Prompt CRTMOD command/s
4.4201	ILE - duplicate member names !
4.4201	RPG/RPGLE with same name in *CCD situation
4.4201	*E72: service pgm not found
4.4201	*SIT mods go to module pool !
4.4201	ILE & OPM program of same name
4.4201	CCD obj deleted on *LIV move
4.4201	Prompt compile TGTRLS *CURRENT
4.4201	Locks on CR cause SAVLIB failure on archive
4.4201	FTP: THNNETMSG logging re-organised
	4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4200 4.4201

3732	4.4201	4.4200: Unable to add obj ovr in CR type ovr
3733	4.4201	Unable to install OPM PGM to *IAC if MOD same name
3769	4.4201	Unable to concurrently develop ILE PGM
3777	4.4201	APYOBJAUT: remove parms REFLIB and REFONLY
3783	4.4201	ASSET: No options when configuring SETS
3936	4.4201	Bug in 4.4103: install same obj in RLS !
3960	4.4201	Array error when positioning to module
3962	4.4201	Cannot find obj override within CR type
3980	4.4201	Can't Mark Delete ILE PGM as *MOD exists
3995	4.4201	Overrides F13 causes WRKAPPCFG crash !
4019	4.4201	Overrides take too long to display
4026	4.4201	Promotion of *GRP *SIT ILE programs !
4029	4.4201	No support for ILE *PGM/*SRVPGM NOSRC !
4036	4.4201	Prevent *MOD delivery on SEE/Chg upgrade
0566	4.4202	After Mark Del of *SIT can still retrieve *SIT
0855	4.4202	CRTCRLIB creates files with default CCSID
2509	4.4202	Archive files being journaled
2516	4.4202	IMPORTLIB – does not support service programs
2694	4.4202	*QMQRY & *QMFORM blank member (UPDSRCREG)
2759	4.4202	IMPORTLIB – allow concurrent development
2812	4.4202	IBM withdrawing support for Office Vision
2815	4.4202	Allow maintenance of Import Register
2849	4.4202	Mark Del *BAS prevents retrieval of *SIT
3050	4.4202	Mark Delete *SIT/*BAS conflict
3263	4.4202	CRTSRCPF with correct CCSID

3440	4.4202	Locked DSPF's not reverted – no warning
3454	4.4202	DUPDTAMBRS: journaling error
3459	4.4202	Include data when importing files
3460	4.4202	IMPORTLIB does not support *SIT or *GRP
3540	4.4202	XREF incorrectly deletes program when CR is at *RDY
3673	4.4202	PRGMVTLOG: System code must be entered for CR
3699	4.4202	WRKINVRQS: wrong application description shown
3797	4.4202	Revert allowed even if no archiving
3799	4.4202	LANSA: Import fails file exists in QTEMP
3877	4.4202	Supplemental groups not recognised by WRKCRDEV
3903	4.4202	CRJOBD copy from DTA LIVE lib if no *MDL environment configured
3972	4.4202	IFS path registration crashes on O#857C for some special authorities
4037	4.4202	No conversion from RPGREF to RPGLEREF
4082	4.4202	External OMT0495 CRTSRCPF command not used
4367	4.4202	Live file locks when compiling for *RDV or *RLS
4373	4.4202	Support for CPP (C++) & HPP objects?
4387	4.4202	Authorisation required for *RDV on TAL's
4391	4.4202	Mark Delete *BAS prevents retrieval of *GRP
4394	4.4202	Mark Del of SQLRPGLE_S shows no related parts
4398	4.4202	Copy against a *GRP CR defaults to *BAS
4417	4.4202	*E99 on source move to *LIV when CCSIDs differ
4418	4.4202	Support for Double Byte Character Sets
4429	4.4202	After 4.4201 APYOBJAUT requires OBJATR
4440	4.4202	CHKCR fails with object type of RPT
4442	4.4202	Cannot retrieve existing SAVF into a CR

4459	4.4202	*E61 error on MOVCR of ILE program
4475	4.4202	Large program rebinds fail with CPD0013 at 4.4201
4640	4.4300	No text on LSTCHGRQS *FULL rpt at 4.4202
4621	4.4300	Tgt Lib not found for *DEL movement at production
4620	4.4300	RPG->RPGLE Conversion Mark Delete Fails
4608	4.4300	V5R2 - RPGLE opcodes ELSEIF,FOR,MONITOR
4606	4.4300	CPF4028 displayed on WRKCHGRQS Opt 46
4602	4.4300	IMPORT not handling LF-NOSRC correctly
4601	4.4300	XIM rcd lock on REMOVE CR OBJECT
4600	4.4300	IMPORTLIB and WRKIMPREG to support SRCT(*CR)
4594	4.4300	MDLs lost on MVT to *live on PROD
4563	4.4300	No ILE parts in SCDM post CVTOMSDTA
4556	4.4300	Conflict of objects of same type with different attribute
4548	4.4300	Automatic release handling fails for *LIV movement
4547	4.4300	DSTCFG fails (CPF2869) if XAS is empty
4545	4.4300	ILE BND pgms overwritten at production
4544	4.4300	On delete of IR - 'Parameter TYPE reqd'
4543	4.4300	Allow direct conversion SQLRPG to SQLRPGLE
4542	4.4300	IFS Attach leads to API error with RC=-1
4540	4.4300	IMPORTLIB ends immediately with no action
4538	4.4300	@LOG gets overwritten by DSTCFG
4519	4.4300	CR library empty on *RDV after Err/Live
4514	4.4300	Incorrect release override is applied
4494	4.4300	Record lock with opt 8 in WRKCHGRQS
4493	4.4300	DSTCFG fails with a CPF2965 or CPF2969

4489	4.4300	*PAGSEG Object type fails in release bld
4482	4.4300	RTVCROBJ does not support GRP/SIT levels
4469	4.4300	*BDR on text field of XREF mbr in RLS
4461	4.4300	FTP TFR shows err but SAVF arrives OK
3637	4.4300	Q/RIs status not removed !
3312	4.4300	*E80 on promotion of a JOBQ
3223	4.4300	CHKCR fails import outstanding
0692	4.4300	APYOBJAUT fails if revoking *ALL on OUTQ
4816	4.4301	OMSJOBD defaults to INQMSGRPY(*SYSRPYL)
4814	4.4301	No support for DDMF file objects
4811	4.4301	Rtv *BAS part for CCD when *SIT part is marked delete
4806	4.4301	Transfer libraries left on production systems
4792	4.4301	IMPORTLIB not incrementing version No's for *BAS
4791	4.4301	Work libs left following various movement scenarios
4775	4.4301	Promote/Rdv of PGM in CR looses prv X-Ref vsn
4753	4.4301	Failure on UIM deletes previous archive
4751	4.4301	OMX/CPL Compiler directives ignored in XRef compiles
4746	4.4301	CRTSRVPGM gives NOT_IN_REG for module libraries
4745	4.4301	File XMB not Cleared in development environment
4742	4.4301	IMPORTLIB to support SQL objects following 4.4300
4735	4.4301	Part transfer to CR at Rdy/Release without authority
4734	4.4301	Multiple versioning warning in WRKAPPCFG
4729	4.4301	CCSID Issue With Files In IFS Directories
4726	4.4301	UPDPRMDTA locks for 60 seconds then finishes OK
4725	4.4301	APYOBJAUT uses wrong attribute on bespoke XOT

4724	4.4301	4.4300 SQL using CRTSRCPF not CRTOMSSRC
4722	4.4301	Vnnn Documentation in SQLRPGLE_S not working
4721	4.4301	Object type overrides failing on site level configurations
4720	4.4301	WRKAPPCFG Site level overrides F13 causes looping
4719	4.4301	Cannot override Recompile for ILE
4714	4.4301	PCKALLARC command parameter confusion
4713	4.4301	CVTOMSDTA problem on 4.4003 → 4.4202 upgrade
4709	4.4301	CCD superseded part deleted from previous CR
4708	4.4301	WRKRLS – JOBD Override Validation in DMS
4707	4.4301	CHANGE Command loops
4705	4.4301	Develop In Module, PF compile – Improve Validation
4697	4.4301	Support for IFS file types of more than 4 characters
4696	4.4301	Asset V8 interface fails with SEE/Change 4.4201
4695	4.4301	Asset V6 interface fails after upgrade to 4.4202
4692	4.4301	CVTOMSDTA attempt to write duplicate records to XOT
4690	4.4301	CVTOMSDTA problem on 4.4001 → 4.4300 upgrade
4688	4.4301	CVTOMSDTA fails on DLTF of OMSTXT*
4681	4.4301	RNQ0106 on F6 to create new ILE program
4679	4.4301	APYOBJAUT fails for SRVPGM having non blank attrib
4668	4.4301	Library restore conflict when multiple logical prod sites
4667	4.4301	WRKOBJAUT SQL0502 Cursor C1 already open error
4664	4.4301	Movement on *INTERPRET2 wrong when overridden
4647	4.4301	REFXREF fails when run for *ALL libraries
4605	4.4301	Read trigger on file movement causes problems
4585	4.4301	Duplicate records on XOT when installing a PE

4583	4.4301	Problem with object level overrides on ILE parts
4560	4.4301	X-Ref recompile problem in release packet build
4557	4.4301	DMS override field to have prompt in Release Dist dsp
4534	4.4301	Site level X-Ref compile failure in release packet build
4533	4.4301	BRWXRFDTA cannot prevent module compile
4532	4.4301	Files with triggers crash after MOVCR
4530	4.4301	APYOBJAUT *OBJ level fails on bound ILE programs
4529	4.4301	Error in X-Ref movement does not fail MOVCR
4520	4.4301	PCKALLARC could offer APP(*ALL)
4512	4.4301	O#800 dumps converting RPG to RPGLE_SRC
4499	4.4301	SEE/Change upgrade RCVRLS uses wrong JOBD
4497	4.4301	*E72 on RCVRLS after 44201 upgrade
4492	4.4301	CRDTA causes install because attribute is PF
4477	4.4301	Multi format LF wrong when new PF's added
4448	4.4301	Option to reset CR Rdy/Release flag
4346	4.4301	Override to previous environment fails on *RDV
4018	4.4301	REFXREF falls over because of lock O#CRL
4013	4.4301	No X-Ref cpl of related module having Sev 10 errors
3961	4.4301	Object not deleted from environment on *RDV
3639	4.4301	CPF2963 – FMTOPT(*NOCHK) rqd on file movement
3412	4.4301	PRGCHGDTA locks XCR to itself (open file elsewhere)
3290	4.4301	*E16 error while deleting CPYREF object
3159	4.4301	Allow CR's at *RDY to be *RDV'd to prv environment
3075	4.4301	*PRV compile on X-Ref object is wrong
2851	4.4301	BRWXRFDTA PF-NOSRC fails

2683	4.4301	Journals applied after CPYF on movement
2606	4.4301	RCVRLS – send back error log
829	4.4301	Libraries over 3 years old never expire
635	4.4301	RDV that fails to move files completes OK
550	4.4301	Stamp output spool files with CR/RIs number
4885	4.4302	Xref Object level overrides not distributed to production machines
4884	4.4302	Option 11=Transfer part forgets target CR number
4883	4.4302	Xref object level overrides not being used for XRef movements
4882	4.4302	Empty work libraries left of system after movement to live
4880	4.4302	Work libs containing *PGM's left on system after movement to live
4878	4.4302	Release incorporates wrong version of Xref programs
4876	4.4302	Validation of same object name with different attribute enforces
4874	4.4302	QSNDR Decimal data error in O#160
4862	4.4302	CR goes to Error/Rdv when containing an SQLPROC part
4838	4.4302	REFXREF incorrectly reports E05 errors in batch
4930	4.4303	SEE/Change RCVRLS to 4.4300 fails
4925	4.4303	*RDV Cross Reference recompile failure issue
4920	4.4303	Cross Referencing compiling SQLRPGLE as RPGLE
4919	4.4303	Cross Referencing not resetting before recompiles
4908	4.4303	E05 on REFXREF for no apparent reason
4907	4.4303	Module CCD part deleted when moved from Acceptance to LIV
4906	4.4303	Compile of SQLRPGLE_S deletes RPGLEREF in SCDM
5034	4.4304	ILE Program creation and update execution commands are wrong
4998	4.4304	CVTOMSDTA halts – duplicate record (44303)
4996	4.4304	CRDTA object not available to after program

4990	4.4304	Object authority not applied on *QRYDFN
4963	4.4304	*BEFORE program not running on prod ACP install
4959	4.4304	*E64 in test environment when OPM XRef promoted
4939	4.4304	Object Authority not applied for user defined object type
4923	4.4304	Changing Object Attribute (SRC FILE/LIB)
4848	4.4304	CHKTAL locks for 60 seconds
4844	4.4304	Mvt of ASSET CR Fails (SAVOBJ in batch)
4835	4.4304	O#310 called too many times causes *E83
4810	4.4304	Duplicate records on XIM from ASSET
4788	4.4304	ASSET: CHKCR_TST not testing integrity
4632	4.4304	CRTSRVPGM not remembering Activation Grp
2926	4.4304	ILE program table corrupted
5061	4.4305	Over 50 libraries in CRJOBD causes error
5060	4.4305	DSTCFG overrides general parameter @SDL
5074	4.5000	N/Fb after retrieval of bound SQLRPGLE
5064	4.5000	CHKCR_TST passes with obj having no text
4879	4.5000	CRTCRLIB CRTSRCPF efficiency improvement
4458	4.5000	Message data fields lost on copy of msg
5082	4.5001	No cross reference for ILE modules in module pool library
5085	4.5001	Cross referencing refresh enquiry (WRKREFXREF) shows no details
5092	4.5001	WRKMVTAUT subfile handling failure
5098	4.5001	ADKCPYDFN fails on retrieval in environment with long library list
5101	4.5001	IMPORTLIB fails when non-standard source files used in app config
5102	4.5001	APYOBJAUT does not work for objects that may have multi attributes
5121	4.5002	Source type & qualifier wrong in AS/SET interface for non ILE parts

5109	4.5002	Upgrade overwrites user defined object types
5097	4.5002	Upgrade Overwrites XOM file for user defined object types
5094	4.5002	E40 error when delivering *INTERPRE2 source with overrides
5093	4.5002	No member created on retrieval of new source based part
5091	4.5002	Session or device error in Release Manager
5089	4.5002	Session or device error on 46=Dependants
5088	4.5002	PRGCHGDTA Failing post 4.4300 when job number is < 100000
5081	4.5002	OME4004 in CHKCR for marked deleted module (ILE cvt to hybrid)
5080	4.5002	Version Numbers fix program for corruption caused by HDM 4792
5065	4.5002	CLP> CLLE_SRC conversion doesn't copy source
5048	4.5002	WRKTAL access option from WRKSYSCFG
5042	4.5002	XMV record locked to job on failed promote of ILE *PGM / *SRVPGM
4992	4.5002	RPGLEREF not delivered to Acceptance Test on Production M/C
4986	4.5002	WRKRLS Opt 17 does not work on production if @SDL implemented
4983	4.5002	CPF3178 problem with XREF and modified ILE program
4960	4.5002	/*CPL* statements not handling quotes or uppercase correctly
4958	4.5002	*E04 when 2 or more overrides on RPGREF / *CPYREF source
4955	4.5002	Cross Reference compile uses wrong CPYREF source
4954	4.5002	OME3190 (can't delete archive library) on promote of PF with LF's
4941	4.5002	CHKCR Creates a default member
4929	4.5002	Movement passes even after failed *CPL
4909	4.5002	O#MVL runs once for each site
4902	4.5002	Erroneous OME3190 in OMS440O report
4901	4.5002	Add DSPOBJHST option to BRWXRFDTA
4899	4.5002	Cross Referencing code incorrectly runs on production machine

4894	4.5002	No STRSEU option for MNUCMD object type
4881	4.5002	Level check after reversion of cross reference program change
4869	4.5002	Cross referencing leaves program behind in Module/Integration
4866	4.5002	Cross referencing misinterprets external DTAARA definitions
4865	4.5002	CPYREF types at process sequence 40 is wrong
4864	4.5002	Cross referencing generated "*VARIABLE" entries
4863	4.5002	RNX1021 hard error when creating a CR
4817	4.5002	/*CPL* statement won't be forgotten
4784	4.5002	Unsent CR in release stays at Q/Release
4671	4.5002	Library descriptions to be put on archive save files when packed
4523	4.5002	Cannot create *BEFORE / *AFTER program templates at Dev / *MDL
4386	4.5002	Can initiate transfer to Prod *LIV when rqst sending to Prod *ACP
3485	4.5002	DUPDTAMBRS always sets MAXMBRS to *NOMAX
3476	4.5002	User enrolment to show user description
2992	4.5002	/*CPL* statements limited in length to 256 characters
5176	4.5003	MOVCR job fails with CPF4102
5165	4.5003	MOVCR hard error RNX0100 in EXTXMLTAG
5164	4.5003	O/R objects get deleted after delivery to previous environment (Prod)
5152	4.5003	E72 on OMS440 but movement doesn't fail
5148	4.5003	Enable convert of RPG directly to SQLRPGLE_SRC
5129	4.5003	Include *LIB object type in XOT for APYOBJAUT

13.3.1 HDM 5034: ILE Program commands

Please note the following execution messages have been modified to specify the library and object name substitution variables correctly:

- OMX6003
- OMX6004
- OMX6023
- OMX6024

If you have modified versions of any of these messages, it may be necessary for you to make further corrections. Modified messages are held in the user override message file, OMSMSGU.

13.4 Installation

This section details any special PE considerations. Consider the appropriate notes in this section before applying an upgrade to your system.

13.4.1 4.2001 (CISC)

Must be applied after 4.2000

13.4.2 4.2002 (CISC)

- Must be applied after 4.2001
- Must <u>NOT</u> be applied on OS/400 level V3R1. Use 4.3000 instead for V3R1 systems. Use this release for systems not yet on V3R1.
- Must be applied to all remote production systems unless production system is V3R1

13.4.3 4.3000 (RISC)

- Must be applied after 4.2001
- Must be applied on V3R1 or higher
- Must be applied to all remote production systems. If production system is at V2R3 or earlier then production system should be upgraded to 4.2002.

13.4.4 4.3001 (RISC)

Must be applied after 4.3000

Must be applied to all remote production systems? No

13.4.5 4.3002 (RISC)

- Must be applied to operating system V3R1 or higher
- Must be applied after 4.3000
- Must be applied to all remote production system? Yes

13.4.6 4.3003 (RISC) & 4.2003 (CISC)

- Must be applied after 4.3000 for V3R1 systems or after 4.2002 for CISC systems
- Must be applied to all remote production systems? Yes, if wishing to use delete facility

13.4.7 4.3004 (RISC) & 4.2004 (CISC)

- Must be applied after 4.3003 for V3R1 systems or after 4.2003 for CISC systems
- Must be applied to all remote production systems? Yes, if wishing to use convert facility

13.4.8 4.3005 (RISC) & 4.2005 (CISC)

- 4.2005 must be applied to OS/400 V3R0M5 or earlier
- 4.3005 must be applied to OS/400 V3R1M0 or later
- Must be applied after 4.3004 (RISC) or 4.2004 (CISC)
- Must be applied to all remote production systems? No

13.4.9 4.3006 (RISC)

- All releases are RISC only from hereon
- Must be applied to V3R1 or higher (There is no V2R3 version of this PE)
- Must be applied after 4.3005
- Must be applied to all remote production systems? No, unless wishing to use data queue object type support

13.4.10 4.4000

Must be applied to V3R1 or higher

- Must be applied prior to year 2000
- Must be applied after 4.3005
- Must be applied to all remote production systems? No, unless wishing to use data queue object type support

13.4.11 4.4001

- Must be applied after 4.4000
- Must be applied to all remote production systems? No, unless distributing ILE objects to production machines and are either recompiling or have a delivery type of MOD (Delivery method in application configuration)

13.4.12 4.4002

- Must be applied after 4.4001
- Must be applied to all remote production systems? No
- Installation of this PE requires some preliminary work. See subsection below
- If you distribute ILE programs, please see the subsection below regarding this.

13.4.12.1 4.4002 pre-installation work

Installation of this PE requires some preliminary work. Perform the following steps.

- Restore the PE (O#THN44002) library from the tape onto the AS/400 and then add it to your library list.
- Prompt command O#THN44002/FIX42 and ensure the parameters are correct before executing.
- Ensure that you remove the library O#THN44002 from your library list before continuing.
- Load the PE in the normal way since the library is already installed you can simply submit a job with the following command. Ensure all SEE/Change users are signed off.

RCVRLS SYSM(THN) RLNO(44002) DMSLIB(O#THN44002)

On submitting the command sign off from SEE/Change to avoid any object locks.

13.4.12.2 Job Descriptions for ILE program distribution

If you distribute ILE programs, it now *essential* to have your application job description in your **program** library as well as the data library on all development and production machines. If you have your application set to always recompile this job description will already exist.

If you do not compile on remote machines you must ensure the job description exists. You can either do this manually or create a CR level *BEFORE program with your next delivered CR that will perform the object duplication before the installation commences:

Simply activate line 9200 (RCNT *NE '001') in the *before* program template and after it add the line

```
CRTDUPOBJ OBJ(xx) FROMLIB(&DLIB) OBJTYPE(*JOBD) TOLIB(&PLIB)
```

```
MONMSG MSGID(CPF0000)
```

Where xx = your application job description name (available in application configuration)

Failure to put the application job description in the program library will result in *E61 errors during installation of CR's containing ILE programs.

Do not install this PE to a production system if that system receives ILE programs and recompiles them. For ILE recompilations on production system upgrade to PE 4.4005.

13.4.13 4.4003

- Must be applied to V3R1 or higher
- Must be applied after 4.4002
- Must be applied to all remote production systems? No, unless using JD Edwards interface or overrides
- See ILE subsection below.

13.4.13.1 Notes for ILE users

If you have ILE switched on for any of your applications and are compiling on promotion or using cross referencing set to *FULL, before you use the new functionality of this PE, please ensure that your application module pool library is not specified as your live program library. If you need to change your application configuration, move your modules to the new module pool library and then run BLDILEREF to complete the new configuration.

Do not install this PE to a production system if that system receives ILE programs and recompiles them. For ILE recompilations on production system upgrade to PE 4.4004.

13.4.14	4.4004	
	-	Must be applied to V3R1 or higher
	-	Must be applied after 4.4003
	•	Must be applied to all remote production systems? No
	•	PE's subsequent to 4.4004 will be at V3R2 until further advised
13.4.15	4.4005	
	-	Must be applied to V4R2 or higher
	-	Must be applied after 4.4004
	•	Must be applied to all remote production systems? No
13.4.16	4.4100	
	-	Must be applied to V4R2 or higher
	•	Must be applied after 4.4005
	•	Must be applied to all remote production systems? Yes, if wishing to use FTP for software distribution
13.4.17	4.4101	
	-	Must be applied to V4R2 or higher
	•	Must be applied after 4.4100
	•	Must be applied to all remote production systems? Yes
13.4.18	4.4102	
	-	Must be applied to V4R2 or higher
	•	Must be applied after 4.4100, can be applied to 4.4101
	•	Must be applied to all remote production systems? Yes
13.4.19	4.4103	
	•	Must be applied to V4R2 or higher
	•	Must be applied after 4.4100, can be applied to 4.4101 or 4.4102
	•	Must be applied to all remote production systems? Yes

13.4.20 4.4200

- Must be applied to V4R2 or higher
- Must be applied after 4.4100, can be applied to systems on 4.4101, 4.4102 or 4.4103
- Must be applied to all remote production systems? Yes
- 4.4200 is the last PE that will be issued at V4R4

13.4.21 4.4201

- Must be applied to V4R4 or higher
- Must be applied after 4.4200
- Must be applied to all remote production systems? No

13.4.22 4.4202

- Must be applied to V4R4 or higher
- Must be applied after 4.4200, can be applied to systems on 4.4201
- Must be applied to all remote production sites? No

13.4.23 4.4300

- Must be applied to V5R1 or higher
- Must be applied after 4.4200, can be applied to systems on 4.4201 or 4.4202
- Must be applied to all remote sites? No, unless wishing to distribute SQL objects

13.4.24 4.4301

- Must be applied to V5R1 or higher
- Must be applied after 4.4300
- Must be applied to all remote sites? No, unless wishing to distribute DDMF objects or take advantage of new trigger support or new override flag on production systems. Also if using group / site cross referencing

13.4.25 4.4302

Must be applied to V5R1 or higher
SEE/Change – Supplementary Manual 4.5003

- Must be applied after 4.4301
- Must be applied to all remote sites? No.

13.4.26 4.4303

- Must be applied to V5R1 or higher
- Must be applied after 4.4302
- Must be applied to all remote sites? No.

13.4.27 4.4304

- Must be applied to V5R1 or higher
- Must be applied after 4.4303
- Must be applied to all remote sites? No.

13.4.28 4.4305

- Must be applied to V5R1 or higher
- Must be applied after 4.4304
- Must be applied to all remote sites? Yes, if release forwarding is to be used.

13.4.29 4.5000

- Must be applied to V5R1 or higher
- Must be applied after 4.4300, can be applied to systems on 4.4301 through 4.4305
- Must be applied to all remote sites? No.

13.4.30 4.5001

- Must be applied to V5R1 or higher
- Must be applied after 4.5000
- Must be applied to all remote sites? No.

13.4.31 4.5002

Must be applied to V5R1 or higher

- SEE/Change 4.5002 will run on OS/400 version V5R1M0 or higher. However, the minimum WDSc supported OS level is V5R2M0.
- Must be applied to all remote sites? See compatibility chart.

13.4.32 4.5003

- Must be applied after 4.5002.
- SEE/Change 4.5003 will run on OS/400 version V5R1M0 or higher. However, the minimum WDSc supported OS level is V5R2M0.
- Must be applied to all remote sites? See compatibility chart.

13.5 SEE/Change Compatibility Charts

The following sections contain compatibility charts covering SEE/Change release history 4.4000 to present.

13.5.1 WDSc plug-in compatibility chart

The following chart shows the compatible combinations of SEE/Change's server code, client code and WDSc workbench release.

Server Version	Compatible Client Version	WDSc Version Required
Below 4.5000	No WDSc plug-in available	N/A
4.5000	1.0.7	5.1.2.4
4.5001	1.0.7	5.1.2.4
4.5002	1.0.7	5.1.2.4
4.5003	1.1.4	5.1.2.4

If you attempt to run an incompatible combination, SEE/Change will detect this and the client issues an error message.

13.5.2 Compatibility 4.4000 through to 4.4301

SEE/Change Compatibility Chart

		Development System															
		4.4000	4.4001	4.4002	4.4003	4.4004	4.4005	4.4100	4.4101	4.4102	4.4103	4.4200	4.4201	4.4202	4.4300	4.4301	
	4.4000	Yes	Yes	Yes	2	2,3	2,3	1,2,3,4	1,2,3,4	1,2,3,4	1,2,3,4	U/C	U/C	U/C	U/C	U/C	
em	4.4001	No	Yes	Yes	2	2,3	2,3	1,2,3,4	1,2,3,4	1,2,3,4	1,2,3,4	U/C	U/C	U/C	U/C	U/C	
	4.4002	No	Yes	Yes	2	2,3	2,3	1,2,3,4	1,2,3,4	1,2,3,4	1,2,3,4	U/C	U/C	U/C	U/C	U/C	
	4.4003	No	Yes	Yes	Yes	3	3	3,4	3,4	3,4	3,4	U/C	U/C	U/C	U/C	U/C	
st	4.4004	No	No	No	No	Yes	Yes	4	4	4	4	U/C	U/C	U/C	U/C	U/C	
Š	4.4005	No	No	No	No	Yes	Yes	4	4	4	4	U/C	U/C	U/C	U/C	U/C	
0	4.4100	No	No	No	No	4	4	Yes	Yes	Yes	Yes	5a,6,7	5b,6,7,8	5b,6,7,8,9	U/C	U/C	
5	4.4101	No	No	No	No	4	4	Yes	Yes	Yes	Yes	5a,6,7	5b,6,7,8	5b,6,7,8,9	U/C	U/C	
Ť	4.4102	No	No	No	No	4	4	Yes	Yes	Yes	Yes	5a,6,7	5b,6,7,8	5b,6,7,8,9	U/C	U/C	
Ы	4.4103	No	No	No	No	4	4	Yes	Yes	Yes	Yes	5a,6,7	5b,6,7,8	5b,6,7,8,9	U/C	U/C	
ð	4.4200	No	No	No	No	No	No	No	No	No	No	Yes	8	8,9	8,9,10	8 to 14	
ž	4.4201	No	No	No	No	No	No	No	No	No	No	No	Yes	9	9,10	9 to 14	
Δ	4.4202	No	No	No	No	No	No	No	No	No	No	No	No	Yes	10	10,11,12,13,14	
	4.4300	No	No	No	No	No	No	No	No	No	No	No	No	No	Yes	11,12,13,14	
	4.4301	No	No	No	No	No	No	No	No	No	No	No	No	No	Yes	Yes	
Notes Minimum OS/400 Levels																	
	Note 1: Yes, u	unless distributing	g ILE objects and r	ecompiling on de	livery or with a dis	tribution mode of	MOD (Module).						4.4000	V3R1			
	Note 2: Yes, u	unless utilising JE	Edwards ierface	or waing multiple	overrides.								4.4001	001 V3R1			
	Note 3: Yes, u	unless utilising da	itabase managem	e commands.									4.4002	V3R1			
	Note 4: Yes, u	unless distributing	g via FTP.										4.4003	V3R1			
	Note 5a: Yes,	unless distributir	ng overrides.										4.4004	V3R1			
	Note 5b: Yes,	unless distributir	ng overrides of a t	ype that were not	possible before 4	4200, in which ca	se no.						4.4005	V3R2			
	Note 6: Yes, u	unless using IFS	functionality.										4.4100	V4R2			
	Note 7: Yes, u	unless utilising the	e new ASSET ierta	ace.									4.4101	V4R2			
Note 8: Yes, unless using DMS overrides for RCVRLS.								4.4102	V4R2								
Note 10: Yes, unless using iong passwords. 4.4103 V4R2																	
	Note 11: Yes	unless using DD	ME object types (DE	De of Divie).									4.4200	V4R4			
	Note 12: Yes.	unless using trig	aer processina (fil	e triager support	or trigger program	realignment supp	ort for triagers in	acceptance test s	vstems on produc	tion machines)			4.4202	V4R4			
	Note 13: Yes,	unless using "Ke	ep Members" ove	rride flag	55 , 5	5	55		, ,	-,			4.4300	V5R1			
	Note 14: Yes,	unless using Cro	oss Referencing or	ver site and group	configurations								4.4301	V5R1			
	U/C: Unsuppo	Jnsupported Combination															

Disclaimer: Every effort has been made to ensure accuracy however we cannot take responsibility for any errors caused by using this information

13.5.3 Compatibility 4.4100 through to 4.5000

At release 4.4302 the compatibility chart was rationalised:

SEE/Change Release Compatibility Chart

		Development System													
		4.4100	4.4101	4.4102	4.4103	4.4200	4.4201	4.4202	4.4300	4.4301	4.4302	4.4303	4.4304	4.4305	4.5000
	4.4100	Yes	Yes	Yes	Yes	1a,2,3	1b,2,3,4	1b,2,3,4,5	U/C	U/C	U/C	U/C	U/C	U/C	U/C
	4.4101	Yes	Yes	Yes	Yes	1a,2,3	1b,2,3,4	1b,2,3,4,5	U/C	U/C	U/C	U/C	U/C	U/C	U/C
Ε	4.4102	Yes	Yes	Yes	Yes	1a,2,3	1b,2,3,4	1b,2,3,4,5	U/C	U/C	U/C	U/C	U/C	U/C	U/C
te	4.4103	Yes	Yes	Yes	Yes	1a,2,3	1b,2,3,4	1b,2,3,4,5	U/C	U/C	U/C	U/C	U/C	U/C	U/C
ys.	4.4200	No	No	No	No	Yes	4	4,5	4,5,6	4,5,6,7,8,9,10	4,5,6,7,8,9,10	4,5,6,7,8,9,10	4,5,6,7,8,9,10	4,5,6,7,8,9,10,11	U/C
Ś	4.4201	No	No	No	No	No	Yes	5	5,6	5,6,7,8,9,10	5,6,7,8,9,10	5,6,7,8,9,10	5,6,7,8,9,10	5,6,7,8,9,10,11	U/C
C	4.4202	No	No	No	No	No	No	Yes	6	6,7,8,9,10	6,7,8,9,10	6,7,8,9,10	6,7,8,9,10	6,7,8,9,10,11	U/C
. <u>e</u>	4.4300	No	No	No	No	No	No	No	Yes	7,8,9,10	7,8,9,10	7,8,9,10	7,8,9,10	7,8,9,10,11	7,8,9,10,11
ū	4.4301	No	No	No	No	No	No	No	No	Yes	Yes	Yes	Yes	11	11
q	4.4302	No	No	No	No	No	No	No	No	No	Yes	Yes	Yes	11	11
õ	4.4303	No	No	No	No	No	No	No	No	No	No	Yes	Yes	11	11
5	4.4304	No	No	No	No	No	No	No	No	No	No	No	Yes	11	11
	4.4305	No	No	No	No	No	No	No	No	No	No	No	No	Yes	Yes
	4.5000	No	No	No	No	No	No	No	No	No	No	No	No	No	Yes

<u>Notes</u>	Minimum OS	400 Levels
Note 1a: Yes, unless distributing overrides.	4.4100	V4R2
Note 1b: Yes, unless distributing overrides of a type that were not possible before 4.4200, in which case no.	4.4101	V4R2
Note 2: Yes, unless using IFS functionality.	4.4102	V4R2
Note 3: Yes, unless utilising the new ASSET ierface.	4.4103	V4R2
Note 4: Yes, unless using DMS overrides for RCVRLS.	4.4200	V4R2
Note 5: Yes, unless using long passwords.	4.4201	V4R4
Note 6: Yes, unless using SQL object types (DDL or DML).	4.4202	V4R4
Note 7: Yes, unless using DDMF object type.	4.4300	V5R1
Note 8: Yes, unless using trigger processing (file trigger support or trigger program realignment support for triggers in acceptance test systems on production machines)	4.4301	V5R1
Note 9: Yes, unless using "Keep Members" override flag	4.4302	V5R1
Note 10: Yes, unless using Cross Referencing over site and group configurations	4.4303	V5R1
Note 11: Yes, unless using Release Forwarding	4.4304	V5R1
U/C: Unsupported Combination	4.4305	V5R1
	4.5000	V5R1 or V5R2 if using WDSc plug-in

Disclaimer: Every effort has been made to ensure accuracy however we cannot take responsibility for any errors caused by using this information

13.5.4 Compatibility 4.4200 through to 4.5003

At release 4.5001 the compatibility chart was rationalised:

SEE/Change Release Compatibility Chart

		Development System												
		4.4200	4.4201	4.4202	4.4300	4.4301	4.4302	4.4303	4.4304	4.4305	4.5000	4.5001	4.5002	4.5003
	4.4200	Yes	4	4,5	4,5,6	4,5,6,7,8,9,10	4,5,6,7,8,9,10	4,5,6,7,8,9,10	4,5,6,7,8,9,10	4,5,6,7,8,9,10,11	U/C	U/C	U/C	U/C
c	4.4201	No	Yes	5	5,6	5,6,7,8,9,10	5,6,7,8,9,10	5,6,7,8,9,10	5,6,7,8,9,10	5,6,7,8,9,10,11	U/C	U/C	U/C	U/C
el	4.4202	No	No	Yes	6	6,7,8,9,10	6,7,8,9,10	6,7,8,9,10	6,7,8,9,10	6,7,8,9,10,11	U/C	U/C	U/C	U/C
s	4.4300	No	No	No	Yes	7,8,9,10	7,8,9,10	7,8,9,10	7,8,9,10	7,8,9,10,11	7,8,9,10,11	7,8,9,10,11,12	7,8,9,10,11,12	7,8,9,10,11,12,13
Š	4.4301	No	No	No	No	Yes	Yes	Yes	Yes	11	11	11,12	11,12	11,12,13
2	4.4302	No	No	No	No	No	Yes	Yes	Yes	11	11	11,12	11,12	11,12,13
2	4.4303	No	No	No	No	No	No	Yes	Yes	11	11	11,12	11,12	11,12,13
핏	4.4304	No	No	No	No	No	No	No	Yes	11	11	11,12	11,12	11,12,13
ы	4.4305	No	No	No	No	No	No	No	No	Yes	Yes	Yes	12	12,13
p	4.5000	No	No	No	No	No	No	No	No	No	Yes	Yes	12	12,13
2	4.5001	No	No	No	No	No	No	No	No	No	No	Yes	12	12,13
	4.5002	No	No	No	No	No	No	No	No	No	No	No	Yes	13
	4.5003	No	No	No	No	No	No	No	No	No	No	No	No	Yes

<u>Notes</u>	Minimum OS	S/400 Levels
Note 1a: Yes, unless distributing overrides.	4.4100	V4R2
Note 1b: Yes, unless distributing overrides of a type that were not possible before 4.4200, in which case no.	4.4101	V4R2
Note 2: Yes, unless using IFS functionality.	4.4102	V4R2
Note 3: Yes, unless utilising the new ASSET ierface.	4.4103	V4R2
Note 4: Yes, unless using DMS overrides for RCVRLS.	4.4200	V4R2
Note 5: Yes, unless using long passwords.	4.4201	V4R4
Note 6: Yes, unless using SQL object types (DDL or DML).	4.4202	V4R4
Note 7: Yes, unless using DDMF object type.	4.4300	V5R1
Note 8: Yes, unless using trigger processing (file trigger support or trigger program realignment support for triggers in acceptance test systems on production machines)	4.4301	V5R1
Note 9: Yes, unless using "Keep Members" override flag	4.4302	V5R1
Note 10: Yes, unless using Cross Referencing over site and group configurations	4.4303	V5R1
Note 11: Yes, unless using Release Forwarding	4.4304	V5R1
Note 12: Yes, unless wishing to use the new file attribute support or the new ILE remote delivery mechanism	4.4305	V5R1
Note 13: Yes, unless wishing to use stream file support distribution	4.5000	V5R1 or V5R2 if using WDSc plug-in
U/C: Unsupported Combination	4.5001	V5R1 or V5R2 if using WDSc plug-in
	4.5002	V5R1 or V5R2 if using WDSc plug-in
	4.5003	V5R1 or V5R2 if using WDSc plug-in

Disclaimer: Every effort has been made to ensure accuracy however we cannot take responsibility for any errors caused by using this information.

13.6 CISC / RISC Considerations

During the CISC / RISC period Thenon released a new RISC based version of SEE/Change (4.3) and continuing to maintain a CISC based version (4.2) for a period. The following diagrammatic explains the history of these releases:

CISC	RISC
4.2000	
4.2001	
4.2002	4.3000
	4.3001
	4.3002
4.2003	4.3003
4.2004	4.3004
4.2005	4.3005
	4.3006

- 4.3000 was largely concerned with the introduction of ILE;
- 4.2002 Included only the elements of 4.3000 that were applicable to CISC machines;
- 4.3001 and 4.3002 were RISC only PTF releases for sequential application to 4.3000;
- 4.2003, 4.2004 and 4.2005 are functional enhancements that were developed in both the CISC and RISC environments. The table above shows the corresponding RISC release;
- 4.2005 was the last CISC release of SEE/Change to be issued;
- 4.3006 was the first RISC only release of SEE/Change to be issued.

13.7 PE Delivery Change notification

13.7.1 Changes to SEE/Change PE delivery (4.4100).

We are changing the conventions we have for supporting SEE/Change. Until now we have stated that there will generally be one software release a year. We changing this method in order to improve flexibility and responsiveness to support issues.

From now on there will be two types of PE. A functional PE containing new functionality such as this one, and a PE containing only program tempory fixes (PTF's).

The PTF releases will normally be small, contain no database changes and will have a low impact for installation. They will be available on the Thenon Web site. All fixes will be delivered in this way, even if the PE contains just one changed object. The frequency of PTF releases will be comenserate with the urgency of the fixes contained within.

Functional releases will contain normally one major item of new functionality like this one, and will be available when complete.

A revised naming convention has been introduced as follows.



It will still be necessary to install all intermediate releases, i.e. to upgrade from 4.4100 to 4.4200 will require the installation of all issued PTF releases between them, 4.4101, 4.4102 etc. There is no concept of a cumulative release.

13.7.2 How this affects you

If you suspect you have a problem with your SEE/Change system the first option should be to check the available PTF's either via the web site or calling support directly. If the fault is not documented in an available PTF release then the issue should be raised with support.

13.8 User Exit Programs

This section lists SEE/Change user exit programs that are build within SEE/Change.

Program Name	Description
--------------	-------------

SEE/Change – Supplementary Manual 4.5003

O#BEXT	OMS550
O#SEXT	OMS100C
O#MEXT	RCVRLS
O#LEXT	CRTCRLIB
O#AEXT	OMSCRTLIB
O#CEXT	WRKCHGRQS
O#IEXT	WRKINVRQS
O#DEXT	WRKOMSDOC

14 Index

*AFTER Processing for automated release handling, 37 @AIE, 215 (*a*,*AUD*, 35 @CAS, 217 @IFP, 225 @ROV, 37 @SNL, 36 4.4200 Authorise Release Distribution, 35 Automatic Release Handling, 36 ADKLIBL, 214 ASSET *After Program, 218 ADKLIBL, 214 Application Configuration, 215 ASSETDTA, 214 ASSETJOBD, 214 Authorities, 217 Best Practice, 218 Configuration, 214 Library List, 215 Save Library, 214 Subsystem, 214 Usage of interface, 219 ASSETJOBD, 214 Authorisation Release Distribution, 35 Automated Release Handling, 36 configuration FTP, 29

Distribution Authorisation, 35 FTP Host name, 30 Host name, 30 IP address, 30 Long Password Support, 39 **Object Types** ADKBAT, 218 ADKDSP, 218 ADKFIL, 218 ADKFLD, 219 ADKMDL, 219 ADKRPT, 218 ADKSUB, 219 Overrides *BYPASS, 34 *CONFIG, 34 by Object, **33** for CR Type, 31 Overrides for DMS submit RCVRLS, 37, 201 QPWDLVL, 39 Release Distribution Authorisation, 35 SEE/Change Dev site code, 31 Send to Live, 36 SNDCR, 36 **TFRCROBJ** Use with ASSET, 218 V5R1M0 4.1.1 Long Password Support, 39 WRKAPPCFG, 215