EspControl

User Manual

READ ME FIRST

Copyright Reserved ©

This document contains proprietary information that is protected by copyright law. All rights are reserved. No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form or by any means, that is electronic, mechanical, magnetic, optical, chemical, manual or otherwise, in whole or in part, without the prior written consent of Cronus Consulting (Pty) Ltd.

Disclaimer

Cronus Consulting (Pty) Ltd hereby disclaims any and all guarantees and warranties for the correct use and application of the ESP software.

Cronus Consulting (Pty) Ltd reserves the right to revise and make changes to the software and the content of this document from time to time without obligation to notify any person of the changes.

Ownership

The ESP Products are developed by, and is fully owned by Cronus Consulting (Pty) Ltd.

TABLE OF CONTENTS

READ ME FIRST 2

1. INTRODUCTION 6

1.1 PURPOSE 6

1.2 PRE-REQUISITES 6

1.3 PRODUCT OVERVIEW 6

2. MENU OVERVIEW AND FUNCTION SELECTION 14

3.1 ISR CONTROL AND PROFILE FUNCTION OVERVIEW (CC500) 18

3.1.1 CC001 – MAINTAIN CONTROL VARIABLES 18

3.1.2 CC002 – MAINTAIN ISR PROFILES 24

3.1.3 CC010 – ISR CODE MAINTENANCE 32

3.1.4 CC025 – ISR PROFILE ENQUIRY 38

3.1.5 CC300 – SCAN PROGRAM SOURCE-CODE 39

3.1.6 CC330 – SCAN SCL'S FOR TRANSFER 40

3.1.7 CC350 – RUN INITIAL BACKUP ROUTINE 41

3.2 ISR TRANSFER FUNCTION OVERVIEW (CC501) 49

3.2.1 CC030 – ISR REQUEST CAPTURE 49

3.2.2 CC035 – ISR REQUEST AND APPROVAL 52

3.2.3 CC040 – APPROVE ISR REQUEST 53

3.2.4 CC050 – LINK OBJECTS TO AN ISR 55

3.2.5 CC054 – COPY AN ISR 68

3.2.6 CC100 – TRANSFER AN ISR BETWEEN ENVIRONMENTS 70

3.2.7 CC105 – TRANSFER WITH SKIP OF ENVIRONMENT 94

3.2.8 CC250 – SELECT OBJECTS FOR ARCHIVING 95

3.3 ISR MAINTENANCE FUNCTION OVERVIEW (CC502) 99

3.3.1 CC051 – TRANSFER AN OBJECT TO ANOTHER ISR 99

3.3.2 CC060 – UPLOAD APPROVAL ISR FOR TRANSFER 101

3.3.3 CC070 – MARK AN ISR AS COMPLETED 104

3.3.4 CC075 – RESET ISR STATUS 106

3.3.5 CC076 – REMOVE ISR FROM SYSOBJH ERROR 108

3.3.6 CC077 – REMOVE AN ISR FROM SYSTEM MESSAGE ERROR 109

3.4 DISPLAY ISR INFORMATION FUNCTION OVERVIEW (CC503) 110

3.4.1 CC080 – DISPLAY ISR STATUS 110

3.4.2 CC085 – DISPLAY OBJECT DEPENDENTS 111

3.4.3 CC088 – DISPLAY OBJECT HISTORY 114

3.4.4 CC090 – Display Objects Linked to an ISR 117
3.4.5 CC093 – Display X-REF History 119
3.4.6 CC095 – Display ISR History 121
3.4.7 CC097 - Display Object Status 126
3.4.8 CC099 - ISR Status Summary 128

3.5 ISR RETURN AND RESTORE FUNCTION OVERVIEW (CC504) 130
3.5.1 CC053 – Link Objects per ISR for RETURN 130
3.5.2 CC061 – Return Approval for ISR 133
3.5.3 CC102 - Return ISR to Previous Environment 134
3.5.4 CC110 - Restore ISR to Initial Environment 138
3.5.5 CC115 - Restore Object(s) to Initial Environment 141
3.5.6 CC200 - Restore Objects Versions to ANY Environment in ISR 143

APPENDIX A – SCL TRANSFER 156

1. MENU OVERVIEW AND FUNCTION OVERVIEW SHOWING SCL'S 158

3.1 ISR CONTROL AND PROFILE FUNCTION OVERVIEW 159
3.1.1 CC002 – Maintain ISR Profiles 159
3.1.2 CC010 – ISR Code 160
3.1.3 CC330 – Scan SCL's for Transfer 160
3.1.4 CC350 – Run Initial Backup Routine 161

3.2 ISR TRANSFER FUNCTION OVERVIEW 162
3.2.1 CC040 – Approve ISR Request 162
3.2.2 CC050 – Link SCL's to an ISR 163
3.2.3 CC054 – Copy an ISR 166
3.2.4 CC100 – Transfer an ISR between Environments 166
3.2.5 CC105 – Transfer with Skip of Environment 172
3.2.6 CC250 – Select Objects for Archiving 172

3.3 ISR MAINTENANCE FUNCTION OVERVIEW 173
3.3.1 CC051 – Transfer a SCL to another ISR 173
3.3.2 CC076 – Remove ISR from SYSOBJH Error 173
3.3.3 CC077 – Remove ISR from System Message Error 173

3.4 DISPLAY ISR INFORMATION 174
3.4.1 CC085 – Display Object Dependents 174
3.4.2 CC088 – Display Object History 174
3.4.3 CC090 – Display Objects linked to an ISR 174
3.4.4 CC095 – Display ISR History 175

3.4.5 CC088 – DISPLAY OBJECT STATUS 175

3.5 ISR RETURN AND RESTORE FUNCTION OVERVIEW 176

3.5.1 CC102 – RETURN TRANSFER ISR TO PREVIOUS ENVIRONMENT 176

3.5.2 CC110 – RESTORE ISR TO INITIAL PROFILE ENVIRONMENT 176

3.5.3 CC115 – RESTORE OBJECT(S) TO INITIAL PROFILE ENVIRONMENT 177

3.5.4 CC200 – RESTORE OBJECT VERSIONS TO ANY ENV PER ISR 177

1. Introduction

1.1 Purpose

EspControl has been designed to assist the system administrator in the migration, documentation and backup of Natural Objects and SCL's between environments on a LUW platform.

Correct usage of the change control system, EspControl, will ensure that any given module/SCL is only updated by one developer at a time. Depending on the ISR profile and/or security definitions, restrictions may exist or be defined to enable specific users to migrate ISR's to and from master environment.

1.2 Pre-Requisites

- Depending on the SHELL environment elected, you will have to enable "rsh", "remsh" or "ssh" commands on each of the server(s).
- Before the EspControl product is installed ensure that basic remote-copy commands can be executed. Refer to the installation documentation for examples of remote shell commands.

1.3 Product Overview

EspControl provides the following functionality:

- Migration of Natural Objects – GDA, PDA, LDA, MAP, COPYCODE, TEXT, HELPROUTINE, CLASS, ADAPTER (AJAX), SUBROUTINE, SUBPROGRAM, PROGRAM
- If EspBatch is installed in each environment, then SCL's may also be migrated from one environment to the next. All available functions that allow SCL's will have an option for SCL's Multiple backups of Source, Object Code and SCL's per environment as defined by the system administrator
- Initial Backup of ALL objects at install or upgrade time before any migration takes place
- Transfer across physical machines and multiple Natural environments.
- Return transfer for specified objects in the ISR across physical machines and multiple natural environments
- Profiles per ISR where each profile is a subset of the Master Control Variables
- Variable DEVELOPMENT environments where the Initial Environment of a profile path need not necessarily be the actual Development environment. Due to this fact, the DEVELOPMENT, unless true DEVELOPMENT, will be referred to as the INITIAL Environment in the rest of this manual.
- A PATH CHOICE which allows the user to decide on the first migration step of an ISR whether to DOWNLOAD from the MASTER or to miss this step and start directly with the UPLOAD from the Initial Environment or an UPLOAD ONLY option.

- Dependency selection will allow an object and all its dependents to be selected, if source exists in the environment
- Automatic restore capability
- Back out and restore on migrated objects/SCL's
- Updating of the Natural buffer pool
- Source allowed or only Object code per environment
- Where SOURCE is ALLOWED a further option is available to REMOVE the source and object after a successful migration to the next environment.
- STOW or CAT per environment or NONE per environment, where only the source is copied and the object is neither STOWed or CATalogued but only SAVEd
- Optional Backup per environment, if No backup selected, no Restore will be done on back-out.
- Objects can be moved from one library to a different library in each environment, where the library may be specified on either the Profile or linked to the individual object when linked to an ISR. An SCL however will always use the same SCL User in each environment.
- Optional Restore during a migration, where the ISR is stopped, if in error, or continues with other objects if no restore required, creating an Error report for the user to use, to manually fix objects in error.
- User or Group security per Profile
- ISR Security where only the "owner" may approve the ISR and link objects.
- A copy facility where a COMPLETED ISR can be copied to a NEW ISR and the original links will be copied if the object is available.
- The COMMENT that is added to the end of an object during migration may be turned on or off.
- Migrate User can be linked where all migration transactions are handled by this user irrespective of the environment being migrated to. For SCL migration this migration user is not used.
- Source can be unloaded via the SYSOBJH function per environment creating a LUW file for further distribution or manual upload.
- System Error Messages (SYSERR) can be transferred from one environment to the other, where the error messages have been set up in the SYSERR Natural function.
- An Audit Report option may be turned on and off, and if on, an audit report will be printed at the end of each migration move. This audit report is per ISR, but if turned on, will produce an audit report for all ISR's, until turned off again.
- An Audit History option where a full audit per selected start and end dates and environments will be produced, either via a report to the print queue or a .csv file. This audit reflects the following events – Create, Approve, User, Transfer, Return, Restore. This is available for both objects and SCL's.

Master Setup

On installation the System Administrator will specify the Natural environments (these environments can reside on different physical machines) and define one as the master environment. The master environment must be the current Production environment. This environment should be captured as the **LAST** environment in your system. However, it is possible to have more than one Production Environment specified if different fusers reside on the same physical machine or the same fuser, but different application setups. It is therefore possible to have multiple environments for each physical machine. Short paths are also available, meaning that only two environments need be selected in a profile.

Dynamic allocation of EspControl functions and security at function level determine who has access to specific functions, with more specific security options that may be set up on different transaction types.

During installation the System Administrator will be requested to specify whether EspBatch is running or not. If Yes is specified, then SCL migration will be available and will work in the same manner as Object migration.

An APPENDIX A at the end of the manual will detail the screen differences when migrating SCL's, and will explain the manner in which SCL's are to be migrated and the process to do so.

The EspControl System consists of 5 sub-menus:

- ISR Control and Profiles
- ISR Transfer ISR
- ISR Maintenance.
- Display ISR Information
- ISR Return and Restore

Operational overview

An initial backup routine may be executed if required, where the user selects the applicable environments. This backup routine is only run when installing the product (this initial backup MUST run at install time) or an upgrade of the software is being done (initial backup is optional). Since the backups are taken when moving “from” an environment and not backed up before moving “to” an environment, an initial backup will ensure that there will be a backup available if a problem occurs when an object is migrated for the very first time.

An ISR can either be registered by a User or a member of the development team. The ISR contains information about the requested development/enhancement which includes: current date, date required, title and a detail description. The ISR is linked to a particular profile which details the path and migration rules for all objects linked to this ISR. Once an ISR has been captured it must be approved by the responsible person, assigned to a developer and time estimation must be included.

Once an ISR has been approved the required objects can be linked to the ISR for transfer from the master environment, which may differ from one ISR to the next, and depending on whether the ISR must start from the Master or different environments due to different profiles set up. As soon as an object has been linked to an ISR it is no longer available for selection by any other ISR. Once a new ISR has been approved, the linked objects of a completed ISR may be copied to this new ISR, which allows an automated link and not a manual one via the user. This does not mean the object is “locked” and no developers can edit it. Therefore, developers should check if an object is already linked before working on it.

When transferring objects between environments the EspControl system will make a backup (if the backup configuration is set on), of the object AFTER the transfer takes place from a non-production environment and will backup an object BEFORE the migration to the Production (Master) environment. Therefore the DOWNLOAD move from Production will neither backup Production or the environment it is being moved to. Once moving in an UPLOAD direction the backup will be made of the environment being moved FROM. When the environment being moved TO, is the MASTER Environment, the object will be backed up in the FROM as well as the TO environment. After an object has been transferred it is compiled in the new environment (if the compile configuration is set on) and removed from the Natural buffer pool, if the compilation fails the object will be restored, if the Restore Indicator is set to YES and the Backup indicator is set to YES. The object will only be compiled if the profile specifies such. If the transfer is from the Master environment to the Development environment, then the ISR will NOT be restored if the compile fails, but an error report will be produced for the user to investigate. A version number is allocated to each ISR and with each backup in each different environment the same version number will be used throughout.

If the object is new and being moved to the Master, no backup will be taken as obviously the object never existed in the Master Environment. If the Restore Indicator is set to No and the backup fails when moving

from an environment or to the Master, the ISR will NOT abort and restore, but a window will be displayed at the end of the transfer with the option of a report being created which will detail the objects not backed up. All backup parameters are written to the History file and may be viewed in function CC095. In the History of the backup cycle, the object, the library and the environment will be recorded as well as whether a backup is taken or not. If a backup fails due to permissions (and the restore indicator is set to no) and the user wishes to redo the last transfer, the objects may be returned to the previous environment (see below) and then re-transferred to create a correct backup.

The SCL backup cycle differs to the object backup cycle, in that the SCL backup is ALWAYS done before the Import of an SCL to the next environment in the ISR being migrated.

Specified objects in an ISR may be returned to the previous environment (objects selected by user and not entire ISR) for further development. Once complete, the normal transfer of objects may be selected and these same specified objects will be returned to the original environment (before the return), with the new changes.

Users may decide whether or not a Backup should be taken in the migration process. If YES, the ISR continues as normal. If NO, the objects will not be backed up and consequently, if one fails, the object will not be restored.

A standard library per environment per ISR may be set up and this library will be used in the entire migration process via the Profiles function. A library may also be specified per individual object for the migration path. The library setup order will be as follows 1. Input Library per object in Linking function, 2. ISR Library in Profile and then 3. Actual scanned Library per Object. If the Initial Environment is TRUE Development, neither the object nor ISR library may used, as the object must be transferred using the scanned in Natural Library. If the Initial Environment is NOT True Development, then any of the Library options may be used.

SCL’s do not have the ability to change the SCL User across environments. The SCL User will always be the scanned in SCL User that shows in the Inventory list.

Users may request Source NOT to be migrated to the next environment, in this instance only the object code will be migrated to the next environment. If Source is No in Development or the Initial Environment, it will be transferred from the Master, allow the user to make changes and then deleted after a successful migration to the next environment, with only the object code being moved. SCL’s will not use this option.

Users may request the Source and Object to be REMOVED after a successful migration to the next environment. This will leave NO source or object in the “environment from” and will have to be DOWNLOADED again before any migration may take place. In this scenario the object will be “tested for

existence" when moving to the MOVE SOURCE environment and if there, the user may either, reject this object, overwrite in the next environment or STOP the entire ISR. The rejection will only be for the ONE move and will re-test when the ISR is again moved up the line. If the ISR is STOPPED, it will be put in hold, and must first be reset via CC075 before it will be allowed to continue. The rules for testing for existence are if the source code is to be REMOVED from either the TO or FROM environment, but only if the Master control variable is marked as N or M n CC001. SCL's will not use this option.

Users may select to use a "short-path" ISR, which means that the Initial Environment may not be the first environment (Development) selected in the profile. In this scenario, the Initial Environment Index must be marked as to what the first required path of a profile should be. The "Master Environment" of a short-path may also not be the TRUE Master and the ISR will complete once the "Master" has been transferred to.

A "PATH CHOICE" has been catered for, whereby the user may decide whether an UPLOAD, either optional or forced, or DOWNLOAD, must occur in the first step of a transfer. Once the ISR is in the migration process, the path choice may not be amended, and if necessary the RETURN option must be used to re-download the object.

A Migration User has been catered for, whereby all migration routines will use this user-id to do the copying, moving, stowing or compiling, restoring and backup functions. The creation of temporary work files and initial "ssh" etc setup will use the actual user-id and so the user running the ISR will need correct authorisation and remote shell command setup. SCL's should not use a migrate user.

Users may decide NOT to have the default comment with the ISR number added to the object. If the COMMENT = N, then this will not happen. A comment may only be added to Development or a profile with an Initial Environment, other than Development, but where this environment is on the local server.

Functionality is provided whereby an ISR may be "Approved for Upload", depending on the profile, whereby the ISR must first be "Approved" before it can be transferred to the next environment. Each environment may or may not require Approval, depending on how the profile was set up. This function would typically be assigned to either the project leader or user.

Additional objects can be linked to or removed from the ISR at any stage. If an object is removed, the user will be asked whether a restore or not is required. If a restore is selected then all changes will be backed out, if not selected the object will be left as is in the environments already migrated to. An object can be transferred from one ISR to another as long as it is in the "Development" stage i.e. status of 01

The EspControl System provides numerous enquiry screens where the status of an ISR can be monitored. An audit trail of the ISR is updated throughout the EspControl System to track the progress of the ISR. An

audit record per object is provided whereby all changes to the object can be traced back to individual ISR's and their related profiles. History of the path followed per object is available as well. The History function reflects ALL moves, errors and restores, and is to be used for investigation in the event of a failed migration or return. A log file of all shell commands and scripts run during a migration is available in an external LUW file, with the key always being the ISR number. Typically, if there is an error in the ISR, the log file may be interrogated to view the actual commands being used and try and manually recreate the error.

The EspControl System will provide PATH VALIDATION at the start of each migration process. This Path Validation will scan through all the linked objects in the correct status for the current migration step and validate all the LUW Directories necessary for Backups, Compiles and any other Natural utility required for the successful migration to the next environment. This path validation will also check that the linked object actually exists in the specific Natural Library for migration. If the PATH VALIDATION fails, the necessary error paths will be written to the History function, for the user to check and correct and then redo the migration step. An error report will also be available to be printed if required by the user. The path validation, tests for all objects before failing and will produce an error list of all the failed objects. If an ISR fails during the Path Validation step, it will remain in HOLD and must be reset via the reset in hold function before continuing. This is done so that the user can investigate the error before any more work is done on that particular ISR. There is a code VALIDATH that can be set to No which will then bypass the path validation except when migrating to Production. However, in selecting this option the System Administrator must ensure that all directories/paths in each environment specified in CC001 have been created. The Path Validation routine will automatically create a missing directory during the routine. Skipping the path validation will make the ISR run faster but will then fail if the paths do not exist. It is therefore preferable only to switch this off after the initial backup routine, which will then create all directories if necessary, during the path validation.

During installation a temporary directory for trace files is created in $ESPCCTMP (this is set up as $CRONUS/espcctmp initially). This directory is used for all trace files in case of errors or script problems. If no errors occur, then these files are deleted after every migration. If a migration fails, and the logfile, that is created per ISR in the shell script directory (see CC001 for further explanation) is not enough for finding the error, please refer to the trace files in the temporary directory.

See examples of these files below:

```
-rw-rw-r-- 1 gm712    sag    345 Sep 10 18:28 remcheck.log.NOMAP.NSM
-rw-rw-r-- 1 gm712    sag    342 Sep 10 18:28 remcheck.log.TST3.NSP
-rw-rw-r-- 1 gm712    sag    616 Sep 10 18:25 remcomp.log.TST3.NSP
-rw-rw-r-- 1 gm712    sag 225645 Sep 10 18:26 scldet.7
-rw-rw-r-- 1 gm712    sag     33 Sep 10 18:26 sclout.7
-rw-rw-r-- 1 gm712    sag    302 Sep 10 18:26 sclout.7.error
```

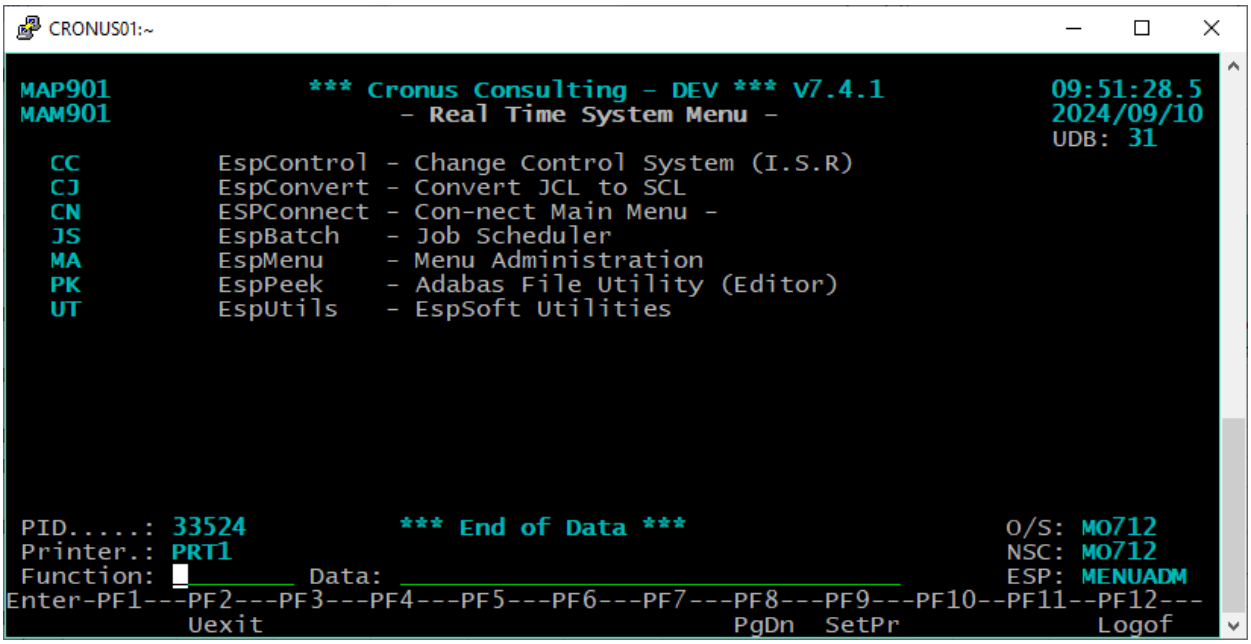
SCL's are copied via Natural routines and not via scripts, as specified for objects above. If any failure occurs during SCL migration, please refer to ISR History function (CC095) to get exact detail of error. The actual SCL detail and errors per SCL is carried across environments in text files, and these will be

available, if code COMPDEBUG is set to Y, by viewing the $ESPCCTMP directory for files with the ISR Number included. The log file in the shell script path will also how the commands used and can be investigated if an error occurs.

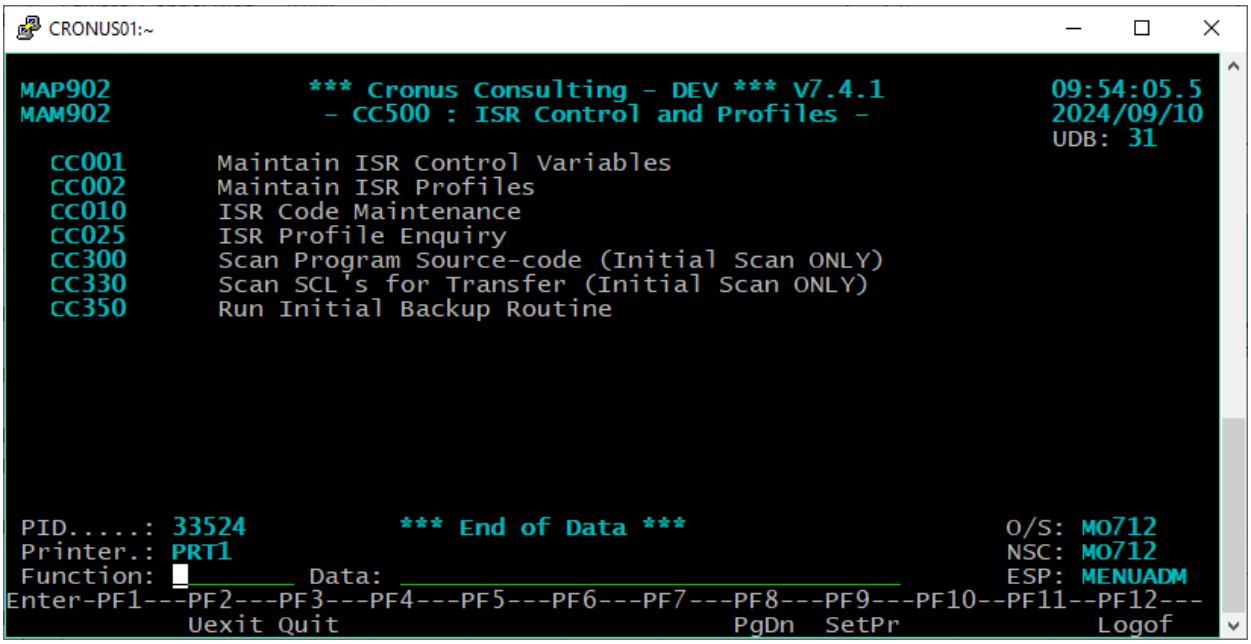
Copyright © 2026 Cronus Consulting

2. Menu Overview And Function Selection

Once a user logs on, the EspControl main menu will be displayed with a list of sub-menu's. These sub-menu's will each display a subset of functions that the user has access to. The following information is displayed on the main menu screen.



Main Menu

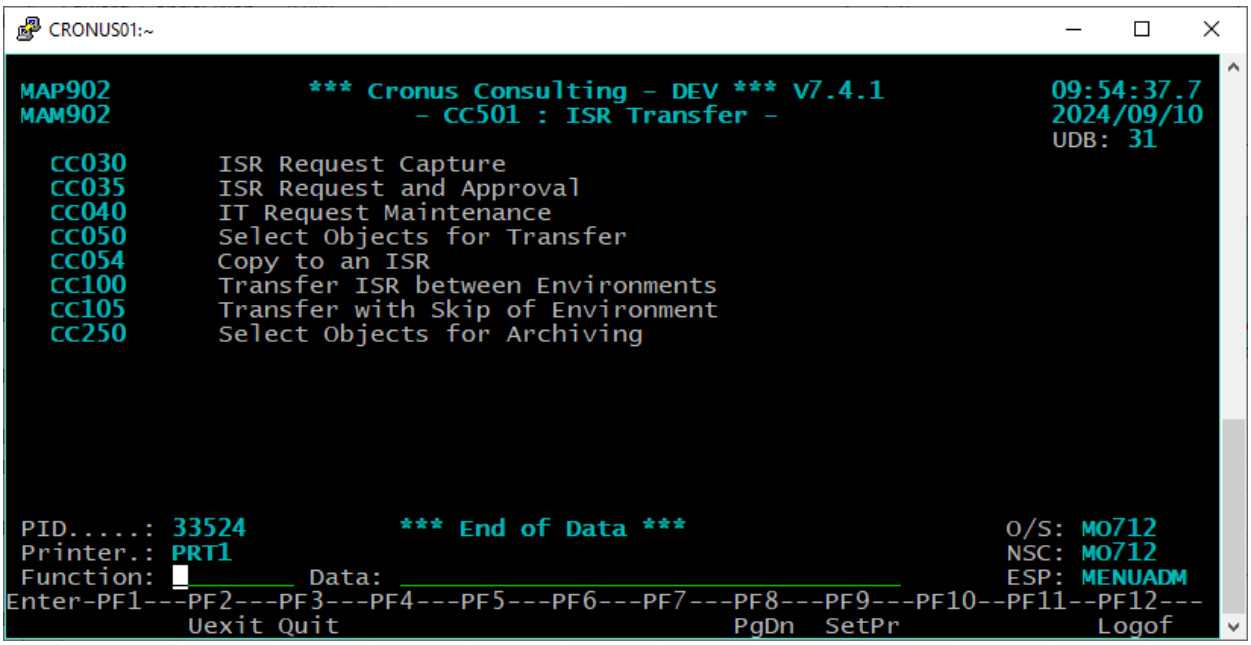


Sub Menu CC500

Function / Menu selection can be done as follows:

- All menu and function selections are cursor sensitive and can be selected by positioning the cursor on the line containing the desired menu or function and then pressing <ENTER>.
- Typing the menu or function name in the function selection area. Parameters required by the function can be typed in the data area, which will be passed to the function.
- The full function for e.g. CC100 may be typed in, or only 100 if required. If only 100 is typed in, the sub-menu that the user is busy with (in this case CC) will be used to complete the function and navigate to the correct function

All sub-menus have specific functions and work alike. See below the rest of the sub-menu's and their related functions. The functions and sub-menus can be allocated in full, or only a portion thereof, to each user or group, to enable further function security in the system. This is done in the MENU ADMINISTRATION section.



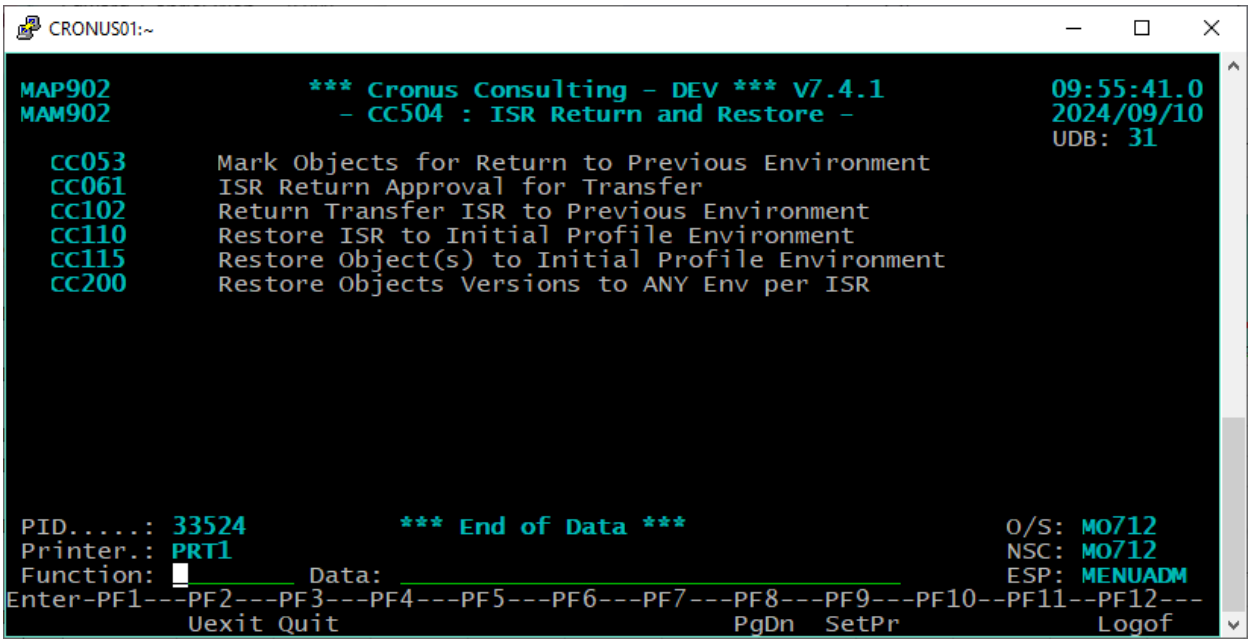
Sub Menu CC501

```
CRONUS01:~
MAP902              *** Cronus Consulting - DEV *** V7.4.1          09:54:58.5
MAM902                   - CC502 : ISR Maintenance -                2024/09/10
                                                                    UDB: 31
  CC051      Transfer an ISR
  CC060      ISR Upload Approval for Transfer
  CC070      Mark an ISR as Completed
  CC075      Remove an ISR from HOLD
  CC076      Remove an ISR from SYSOBJH Error
  CC077      Remove an ISR from System Message Error

PID.....: 33524          *** End of Data ***                    O/S: MO712
Printer.: PRT1                                                  NSC: MO712
Function: _______ Data: ____________________________            ESP: MENUADM
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
           Uexit Quit                          PgDn  SetPr             Logof
```

Sub Menu CC502

```
CRONUS01:~
MAP902              *** Cronus Consulting - DEV *** V7.4.1          09:55:23.8
MAM902                - CC503 : Display ISR Information -           2024/09/10
                                                                    UDB: 31
  CC080      Display ISR Status
  CC085      Display Object Dependants
  CC088      Display Object History
  CC090      Display Objects Linked to an ISR
  CC093      Display X-REF History
  CC095      Display ISR History
  CC097      Display Object Status
  CC099      ISR Status Summary

PID.....: 33524          *** End of Data ***                    O/S: MO712
Printer.: PRT1                                                  NSC: MO712
Function: _______ Data: ____________________________            ESP: MENUADM
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
           Uexit Quit                          PgDn  SetPr             Logof
```

Sub Menu CC503

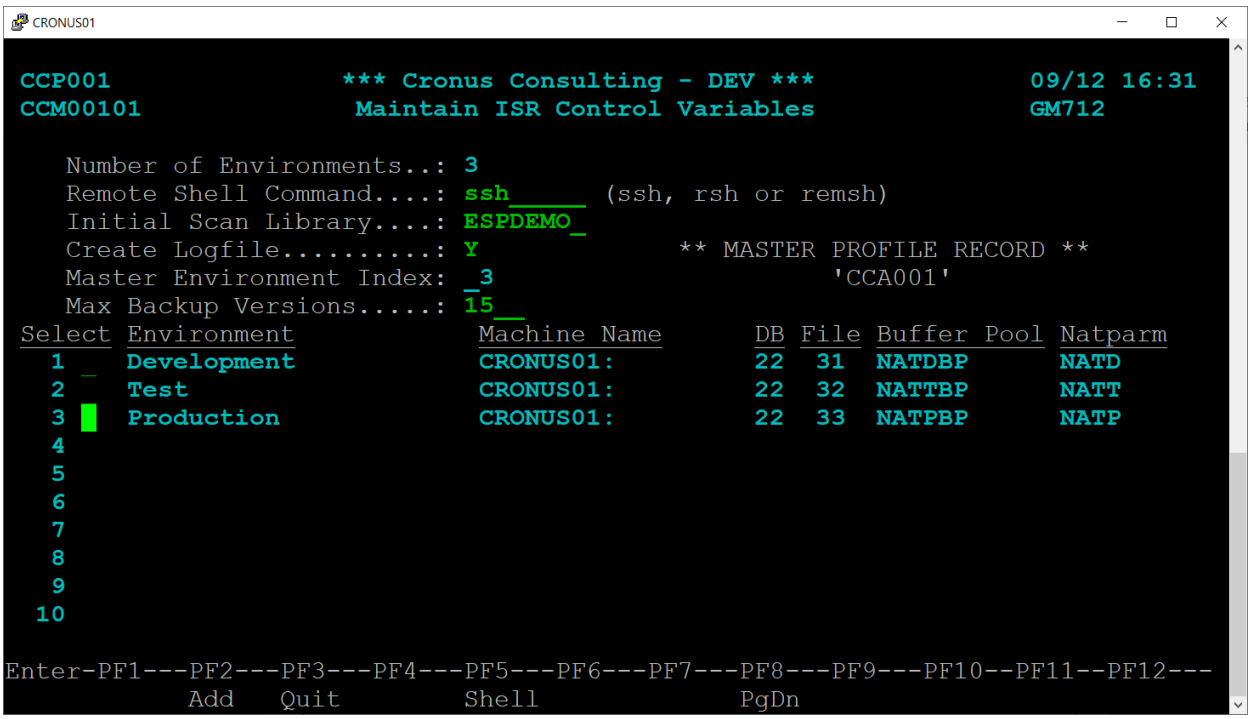


Sub Menu CC504

3.1 ISR Control And Profile Function Overview (CC500)

3.1.1 CC001 – Maintain Control Variables

This function is used to define the EspControl control variables for the Master Profile Record. This is used as a default to create various migration Profiles in CC002. These variables are used during the execution of the migration process and are defined in two groups: a) **Global** variables – these are set once and are applicable for all environments defined in EspControl. b) **Local** variables – definitions for each of the Natural environments that forms part of the EspControl environment. Information regarding the Natural program environments, libraries, host destinations and paths are defined and maintained under the local variable set-up. An environment entry should be defined for each of the Natural program environments that are required to be migrated to. If objects need to be moved to another section of the same environment for e.g. copied from one library to a common library for all applications in the same environment, then this must be set up as a separate entry as well (therefore there may be two entries for Development for example) and then manipulated in the Profile section



ISR Control Variables

This is the default EspControl record (the Master profile) where the number of environments in total, are specified and their related Path data (see next screen image). The following function (CC002) allows the setting up of a profile whose environment path may differ, but the Path data behind each environment is taken from this Master Profile record. An environment, once set up, may never be deleted, if it is no longer required, it must just not be linked to any new Profiles that have been set up. This is because of History and Restore functions.

The following global control variables should be defined:

Number of Environments	:	The total number of Natural environments needed for migration. This will be set up automatically once all individual environments have been captured.
Remote Shell-Command	:	The remote access command protocol to use for transfers between environments. (ssh, rsh or remsh).
Initial Scan Library	:	Default in CC050 and other enquiries reflecting the Object Inventory.
Create Logfile	:	Enable logfile (for error checking this must be set to Y).
Master Env Index	:	The Master Environment (TRUE PRODUCTION). This will default to your last environment.
Max Backup Versions	:	The maximum number of Backups to be taken in each Environment when transferring objects. This version number is the same for each environment move.

The **LOGFILE** should always be marked with Y, as it gives the user command detail and return code detail for all commands issued via EspControl throughout the migration, path validation, SYSOBJH and SYSERR loads, return, backup or restore processes. The log file is ALWAYS found in the first environment specified in CC001 (even if the migration is for e.g. to Production) and is situated in the Shell Script Path directory as indicated in the below screen and then appended with /logs. Should any move through EspControl fail, these log files may be interrogated in LUW file created. Each ISR has its own log file in the shell script path/logs directory with the naming convention ISRnnnnnnnn.LOG where n is the ISR number. Each EspControl routine adds to the log file of an ISR and this is never over-written.

Local control variables for each environment:

```
CRONUS01
CCP001                 *** Cronus Consulting - DEV ***              09/12 16:31
CCM00102                Maintain ISR Control Variables              GM712

Environment .......: 01
Environment Name...: Development_________
Local Indicator....: Y
Machine Name.......: CRONUS01:______
Fuser DBID.........: _22
Fuser File Nr......: _31
Fuser Path.........: /opt/softwareag/Natural/fuser_natd_________________
Fuser Backup Path..: /opt/softwareag/Natural/fuser_natdbck/.BCK________
Source Staging Area: /opt/softwareag/Natural/fuser_natdbck/.SSA________
Nat BufferPool Name: NATDBP__
Natparm Module Name: NATD____
Natural Nucleus....: natural_____________
Allow Source Ind...: Y
Shell Script Path..: /opt/softwareag/cronus/ccont_eco___________________
QA Required Ind....: N

                                Clear all fields?  NO_    (YES or NO)

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
            Add   Quit        Shell             PgDn
```

Control Variables per Environment

The following control variables should be defined:

Environment	: Environment Number (System allocated)
Environment Name	: Environment Name – Description
Local Indicator	: Local indicator to server where EspControl is running
Machine Name	: Machine name/Host-name or IP address
Fuser DBID	: Fuser DBID as defined in NATPARM
Fuser File Number	: Fuser FILE NR as defined in NATPARM
Fuser Path	: Complete path to fuser where source and object exist
Fuser Backup Path	: Complete path to backup fuser
Source Staging Area	: Area used by EspControl to stage the source-code for Allow-Source = N. Even if Allow Source is not N, a copy is moved here so that this variable may be changed in future.
Nat buffer Pool Name	: Natural Bufferpool name for this environment
Natparm Module Name	: Natparm that is used to access this environment
Natural Nucleus Name	: Natural nucleus name (must be in default path)
Allow Source Indicator	: Indicates if source code should be allowed or Moved (used for default as this is specified in the Profile)
Shell Script Path	: The complete path to ESP scripts, log files and transfer (ccont) See below for Development default

QA Required Indicator : Indicates an approval step is required before copy to an Environment.
This is a default only as specified per separate ISR Profile.

The shell script path for Development when CC001 is first set up after install, will default to the $CRONUS/ccont_eco directory, which is where all the EspControl scripts are installed from. This directory should keep this same name for Development and this entire directory must be copied to all environments. If the environments are on the same server, then this directory must be copied with a new name, for example, adding the environment name as a suffix $CRONUS/ccont_eco_dev2. If all environments are NOT on the same servers this name may remain the same. However, for ease of use it is better to include the environment name as a suffix, even if the environments are on different servers. Remember to update these directory names correctly in CC001 for each environment as set up in the install.

The Allow source Indicator and the QA Required indicator are only defaults in the Master CC001 Control record and can be changed in each Profile defined (CC002) depending on requirements. **The Allow-source indicator on the Control record (CC001) determines if OBJECT TESTING occurs between environments and if dependents are allowed.**

Caution: If Natural security is installed, the access to the Natural application system should be defined with **AUTO=ON**

The ISR code, SAMESTAGE, determines if the Staging Area Path may be the same in any of the environments. **Caution**: Remember that if this staging area is allowed to be the same and environments exist in the same fuser, the objects with the same object name and library will be over-written. See CC010.

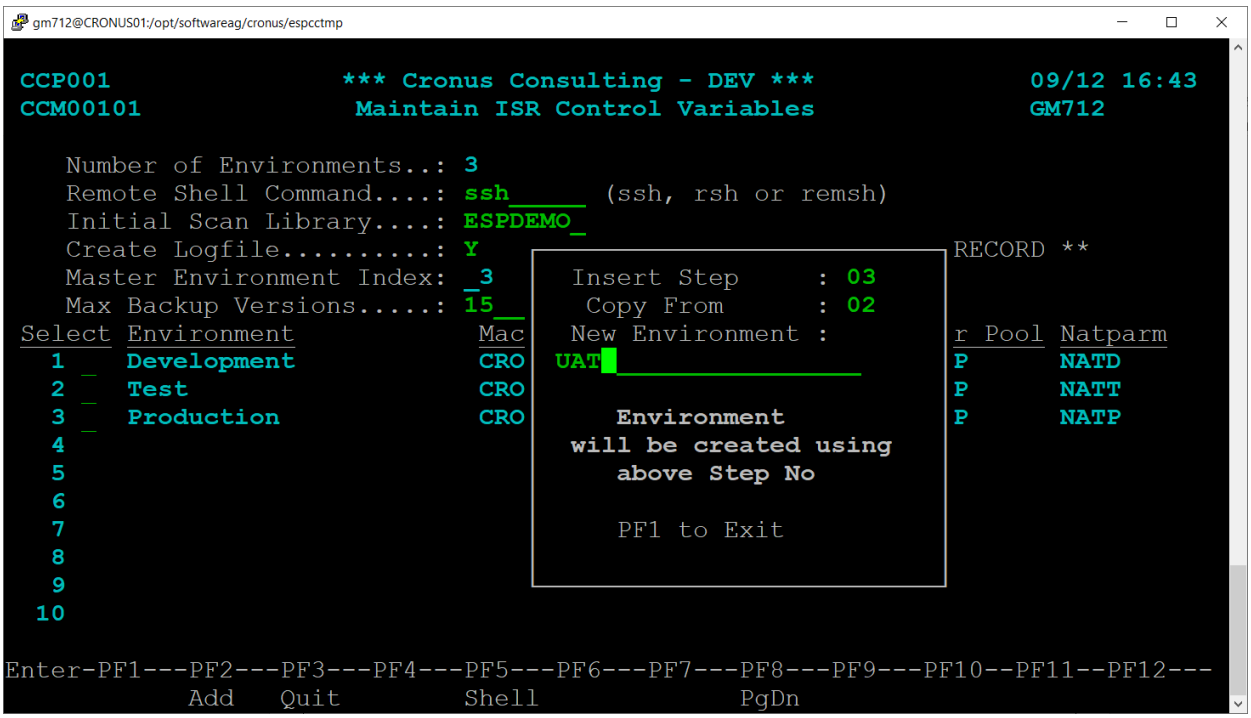
When creating control variables, a **MASTER PROFILE** is automatically created, using the entered local variables and defaults, and this can be enquired on in the ISR PROFILE function (CC002). This Master Profile may not be used to link to ISR's and migrate objects. This profile cannot be modified but may be enquired on. This default profile is named **MASTER.** The MASTER PROFILE is re-created every time a change is made to CC001.
Caution: Any natural library **startup** programs should be limited to start **only** if *DEVICE is **not** set to "BATCH". This causes background tasks in EspControl to **fail**. If **startup** programs are used the default **natural nucleus** command should be change to "**natural batch**" in function CC001. As long as Batch Execution is NOT marked with '**Y'** in Natural Security, if Natural Security is being used.

```
…
Nat BufferPool Name: NATDBP
 Natparm Module Name: NATD
 Natural Nucleus....: natural batch
 …
```

Example where the "natural" command is changed to use "batch mode"

The program **ESPERRTA** is used to **reset** the standard/default ***ERROR-TA** error handling programs for the migration session. This program should exist in library **SYSTEM** in all environments. As part of every migration, it is automatically included in the PATH VALIDATION routine and the ISR will abort if it does not exist in any particular environment in Library SYSTEM.

During the install either the "ssh" or "rsh", depending on the chosen shell, must be authenticated and to test if this is correct, press PF5 from CC001 on the header screen. PF5 may only be tested (when setting CC001 up for the first time), when all variables are entered correctly for the header screen. It is useful to test this before entering any environment variables, as specified below, as if this does not work, then the validation for the paths entered per environment will not work, and return an invalid path, even if the path is valid. Remember that this authentication MUST be done across all servers.



PF2 – Add or Insert Environment

Once CC001 has an environment and the user has exited CC001 and saved the data, any future updating to this function will require the use of PF2. PF2 allows the **insertion** or **addition** of an environment, as seen in the above example. The user may copy all control variables from a specific environment to the newly inserted one. The user must then change the details to the new environment's variables. Please note, until all variables have been amended and SAVED, this new environment will not be added. Use PF1 to exit without completing insert. This "insert" may be done at any time, even after profiles (CC002) have been set up. This function will insert or add a "nil" path to all profiles already created, with a PATH-REQUIRED of NO. New profiles via CC002 may now be set up to use this new environment.

3.1.2 CC002 – Maintain ISR Profiles

```
gm712@CRONUS01:/opt/softwareag/cronus/espcctmp

CCP002                *** Cronus Consulting - DEV ***              09/12 16:44
CCM00201                   Maintain ISR Profiles                   GM712

   Number of Environments..: 3          Profile................: PRD2DEV
   Initial Scan Library....: ESPDEMO_   Profile Description...: PROD TO DEV
   Master Environment Index: _3         Initial Env Index.....: _1
   Restore (Y) or (N)......: Y          Path Choice (Y/N/U/S/C): Y
   Migrate User (optional).: ESPCNTRL   Comment (Y) or (N)....: Y   (N)one
   SCL Type (Y) or (N).....: N  Allow   A-A SrcUnl   User/          (S)tow Back
 Select Path Environment        Source Y/N  Y/N G   Grp    Library (C)at  Y/N
   1 _   Y   Development           Y    N    N  _ GM712___ ________  S     Y
   2 _   Y   Test                  Y    N    N  _ ________ ________  S     Y
   3 _   Y   Production            Y    Y    N  _ ________ ESPDEMO_  S     Y
   4 _   _                         _    _    _  _ ________ ________  _     _
   5 _   _                         _    _    _  _ ________ ________  _     _
   6 _   _                         _    _    _  _ ________ ________  _     _
   7 _   _                         _    _    _  _ ________ ________  _     _
   8 _   _                         _    _    _  _ ________ ________  _     _
   9 _   _                         _    _    _  _ ________ ________  _     _
  10 _   _                         _    _    _  _ ________ ________  _     _
                                                 G-Mark with Y for Group
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                            PgDn
```

ISR Profile Capture

Profiles are required to be set up and a specific profile will be linked to an ISR. These variables are used during the execution of the migration process, using the **actual paths** and Natural Parameters from the Master Profile Record. These are set per environment on the **MASTER Profile in CC001.** The purpose of the separate Profile Record is to allow different ISR's to follow their own path. The environments are reflected as set in the Master Profile and depending on the required PATH, '**Y**' or '**N**' must be entered. If the environment is not selected, the ISR will skip to the next environment whose path = "Y". Any ISR linked to a specific profile will follow the path of "Y", and must include at least two environments.

The first entry into CC002 with an addition of a new profile will automatically reflect defaults as '**Y'** for required path and '**S**" for **STOW** for all required entries up to the Master Environment Index. These may be amended according to user specifications. Profiles may **not** be modified if an **open** ISR **exists** for that **profile**. This is to prevent migrations from going wrong. Once all ISR's linked to a profile have been completed, a profile may be changed. Profiles may NEVER be DELETED. This is because History could exist for this profile and the function CC200 allows restores from old ISR's, even if no more migration occurs with this profile. Profiles may be **Suspended** which will mean that the profile will not be allowed to be linked to any further. Enter "S" under path choice for suspension. An exception exists to the modification of an open ISR, whereby the profile may be modified when it comes to the USERID or GROUP entry on the profile. See below -

```
gm712@CRONUS01:/opt/softwareag/cronus/espcctmp
Open ISR exists, CAN only change USERID or GRP or Delete
 CCP002                 *** Cronus Consulting - DEV ***                 09/12 16:57
 CCM00201                    Maintain ISR Profiles                      GM712

    Number of Environments..: 3          Profile................: PRD2DEV
    Initial Scan Library....: ESPDEMO_   Profile Description...: PROD TO DEV
    Master Environment Index: _3         Initial Env Index.....: _1
    Restore (Y) or (N)......: Y          Path Choice (Y/N/U/S/C): Y
    Migrate User (optional).: ESPCNTRL   Comment (Y) or (N)....: Y   (N)one
    SCL Type (Y) or (N).....: N  Allow   A-A SrcUnl   User/          (S)tow Back
 Select Path Environment         Source Y/N  Y/N G   Grp    Library (C)at  Y/N
   1 _    Y   Development           Y    N    N   GM712___ ________  S     Y
   2 _    Y   Test                  Y    N    N   _ ______ ________  S     Y
   3 _    Y   Production            Y    Y    N   _ ______ ESPDEMO_  S     Y
   4 _                              _    _    _   _ ______ ________  _     _
   5 _    _                         _    _    _   _ ______ ________  _     _
   6 _    _                         _    _    _   _ ______ ________  _     _
   7 _    _                         _    _    _   _ ______ ________  _     _
   8 _    _                         _    _    _   _ ______ ________  _     _
   9 _    _                         _    _    _   _ ______ ________  _     _
  10 _    _                         _    _    _   _ ______ ________  _     _
                                                   G-Mark with Y for Group
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                            PgDn
```

Modification of Open ISR

The following rules apply:

- Path cannot be marked as **'Y'** for any environments greater than the Master Environment Index specified in the profile being worked on.
- The Master Environment index is the main control where the path of the selected ISR will be **started for the download** of existing objects and will in turn complete an ISR cycle.
- The **first** Environment of the Master Profile on the Control Record CC001 (but not necessarily the user profile) is always the "Development" Environment where all new objects must be created and be scanned in CC050, with the exception of multiple Development environments that may exist on the same local server. Whenever objects are migrated to the "development" environment they will always be downloaded using the library of the object in the Inventory list (CC050). Different libraries may be selected per environment (in CC050 or Library in CC002), with the exception of Development. All new objects will start their migration path from this Development environment. If more than one environment exists in the "Development Area", an object may be scanned in to this environment as well, by entering the Environment Index in CC050, as long as these "Development environments" have different library names but exist in the "Development" fuser. Having more than one Development area defined in CC001, requires the use of different libraries to avoid code being overwritten.
- Environment number 01 in CC001 is referred to as the True Development environment, but may be left out of a path profile. If the Development environment is to be ignored, then the **INITIAL ENVIRONMENT INDEX** must be marked with the environment number that will form the first

environment in the transfer of the objects. This scenario is only normally used for Emergency type profiles. A normal profile will always include the Development environment.

- If the Initial Env Index is not 1, then all paths less than this environment number MUST be marked with an **'N'.**
- **PATH** must be '**Y**' or '**N**' and must be '**Y**' for Initial Environment and Master Index Environment marked on that specific profile.
- **ALLOW SOURCE** must be '**Y**' or '**N**' or **'M'** however it must be 'N' if the Path is set to 'N'. If Allow-Source = 'N', only the object code will be migrated. If ALLOW-SOURCE = 'N", then the staging area defined in the Master Profile is used for keeping the source. If Allow source is set to M, then both the source and object will be migrated to the next environment and then deleted from the "From Environment". If ALLOW-SOURCE of either the FROM or TO environment is set to M, a further "object" check will be applicable. However, this check will only come on if this indicator has been marked as such on CC001 as well. Therefore if only a specific profile is marked with 'M', then this check will not be done.This check will inform the user if the object already exists in the TO environment. The user will then have a choice to overwrite the object in the TO environment or REJECT the object from being migrated to the TO environment, or alternatively, STOP the ISR from continuing until the user has sorted out the problem. If stopped then when necessary, RESET the ISR from HOLD and continue with the TRANSFER. The Staging Area is always used in the background when migrating objects or restoring source of an object, even if source is allowed. This is to allow the user the option of changing profiles from Allow-source of YES to NO, as there will always be a last version of the source residing in the staging area. Therefore, if source is ever restored, the source is sent to both the normal fuser and the staging area in order to keep objects in synch.
- **TEXT and COPYCODE** - If text or copy-code is to be migrated, if the ISR has an Allow-Source of NO, because a text or copy-code only has source, the source will be migrated. If a restore is needed, the same rule will be applicable, restoring object only for object types other than text or copy-code, and source for text and copy-code.
- **UPLOAD APPROVAL** must be 'Y' or 'N'. However, it must be 'N' if Path is set to 'N'. If 'Y' is chosen then a user will be required to approve the upload TO that particular environment before migration will be allowed. This indicator will not be used when moving Down from the Master, but if the ISR is used for Upload only, it will request approval for the first migration move, if marked with a Y.
- **USERID** must be a valid User (and not a group). The field need only be entered if security is required around the actual migration function (CC100). If left blank, the transaction security will be used. i.e. if CC100 allocated to a user's menu, the function will be allowed, but if a user-id has been added to an environment on a profile, then this will be a further check and only the allocated environment user to migrate FROM the environment. An error will be displayed if a user-id has been allocated and another user attempts the migration. Note, if a user must have security to be the only user to move TO for

example, Production, then the user-id must be allocated to the environment BEFORE production, as the check works in the FROM environment.

- **GROUP** must be a valid Group (and not a user). The Group security works the same way as explained in the USERID section directly above. However, the user attempting the migration must from part of the Group linked to the environment. Note, both the **USERID** and **GROUP** are entered in the same column, but if it is a group that has been entered, mark **column G with a Y**. Only the userid and group column may be amended when an open ISR exists against the modified profile.
- **STOW/CAT** must be '**S'** for stowing or a '**C'** for catalog or **'N'** for NONE meaning a move only. The STOW option will ensure that both the SAVE and OBJECT dates are updated. The CATALOG option will only update the object date and the SAVE date will remain. The 'True Production' Environment is set at 'S' or 'C' and cannot be changed to an 'N', unless the code NOSTOW (CC010) has been set to a 'Y'. It must be noted that if objects have to be restored, both the SAVE and OBJECT date will reflect the restore date, unless function **CC200** is used as this **ONLY** restores source and must be manually Stow'ed. The NONE option will transfer the code and do a SAVE only. This will update the save date and will reflect NO dates for the object code or any object code itself. If the object is moved to an environment, where the source and object already exists, this old object code will be removed via an UNCATALOG, so that only source remains. Using the restore function, these same parameters will be used when restoring source or object.
- **BACKUP** must be 'Y' or 'N'. The 'True Development' Environment is set to 'Y' and cannot be amended. If Backup is marked as 'Y' migration will proceed as normal, if marked as 'N", no backup of the object will be done and consequently no RESTORE will be done if a problem occurs and objects must be restored, or if the Restore functions are selected from the menu. Backup must be 'N' if the Path is set at 'N'.

  The backup function follows the following path depending on the profile linked to the migrated ISR –

  1. If the Master Environment is not TRUE PRODUCTION, a backup of the object is taken as it is downloaded to the initial environment of the profile.
  2. If the Master Environment IS equal to true production, a backup is NOT taken when the object is downloaded to the initial environment.
  3. A backup is NOT taken of the object in the environment being migrated TO.
  4. Once the object has been maintained by the user, or a new object has been developed and a migration is now in progress, a backup is taken of the object in FROM environment as it is being migrated to the next environment.
  5. If the ISR is being migrated to the Master environment: a). if the Master is NOT True Production, NO backup is taken of the object in the TO environment as this has already been done when downloading occurred, or b). if the Master IS True Production, then a backup is taken of the object in the TO environment before migration occurs and over-writes the object. In scenario b) both a FROM and TO backup will take place.

6. An “F” for From and “T” for To will be displayed on the screen for the user to view if a from or to backup is being processed when migrating via CC100. This is also viewable in the History function (CC095).
7. During a migration, if any backup is unsuccessful and the restore indicator is set to N, then an **E** for Error will be displayed under the backup column of the migration screen, but will NOT ABORT the ISR migration. At the end of the migration process a non-backup error report will be sent to the printer for the user to view. If the backup is not successful, then the object may be returned via CC102, the backup error fixed, and then retransferred via CC100 for a backup to be made. This sequence of non-backups can also be viewed in the ISR History enquiry (CC095).

- **RESTORE IND** must be ‘Y’ or ‘N’. If ‘Y’ and if any object in the migration has a problem, the migration will stop and the entire ISR will be restored depending on the Backup Indicator and restored to the version that was backed up as specified above. If ‘N’ and any object in the migration has a problem, it will be bypassed and the other objects in that ISR will carry on with the migration. At the end of the migration, the user can request an Error report to be printed, which will reflect all error objects. This sequence of error events can also be viewed in the ISR History enquiry (CC095). During a migration, if any object is unsuccessful, then an **E** for Error will be displayed under the backup, copy or compile column of the migration screen in CC100, depending on what section is in error. If an ISR is restored and no backup yet exists for the object being restored, the object will be restored to the previous backup version of that object i.e. the last backup taken in the last migration of that object, no matter the ISR number. If a backup in that ISR does exist, this backup will be used. View the History function if a restore tales place to see which backup version was used.
- **ISR LIBRARY** – This ISR library will replace the library of the object in the CC050 when migrating to an environment. However, the ISR Library is not allowed for the ‘Development’ Environment, as the scanning of the source code determines the library for ‘Development’. From the second Environment onwards, this ISR Library can be different to the scanned library of the source code, and this ISR Library will be used in the move from one environment to the next. If a copy of the source code is required in ‘Development’ in another library to the scanned one, use PF1 in transaction CC050 to do a duplicate copy to another library. In this way, a different library to the scanned one can be used in ‘Development’. This ISR LIBRARY will however, be overridden by the Object Library that can be specified in transaction CC050 per object per environment. The ISR Library should be used when the entire ISR should be migrated to a specific library and the Object Library should be used when objects in an ISR must be migrated to a different library to the scanned ones. This Library must be valid and the function will return an error message if an invalid library is entered. If left blank, the migration will use the default “scanned” library.
- **“GENERIC” ISR LIBRARY –** an ISR LIBRARY may be specified as GENERIC instead of the actual library in the environment. This GENERIC library is controlled by the site and needs to have coding added to the user exit, **ISNUX100**, that exists in library ESPSOFT. The instructions for this coding reside

in ISNUX100 itself. The Generic library should be used if the library name differs from one environment to the next or perhaps from one object to the next or from one developer to the next. The coding added to the user exit must specify the rules to which the library name will be changed automatically, and in this manner will become the “ISR LIBRARY” through all migration routines. If the GENERIC library is not to be used, leave ISNUX100 as it is. ISNUX100 coding may be used in conjunction with the codes under CODE SYSTEM in CC010.

NOTE: When setting up the coding in ISNUX100, the field #TEMP-LIB must always be populated and the following statement is mandatory, once it has been populated –

MOVE #TEMP-LIB TO #TRANSLATE-LIB(#XX)

- **SRC UNL – SOURCE UNLOAD –** This indicator is to be set if the user requires the migrating source to be unloaded as a SYSOBJH file at the end of a successful migration to a particular environment. If Unload is required mark with ‘Y’ else mark with ‘N’. If ALLOW-SOURCE = ‘N’ in any environment other than Development, the unload of the source will not be allowed. Due to the fact that Development always has source even if ALLOW-SOURCE = N, until a successful migration to the next environment, it is allowed here. The SYSOBJ function to unload the source is called once all objects have been successfully migrated or if NO RESTORE is required and the SRC UNL Indicator has been marked with a ‘Y’. For e.g., if environment 1 is marked, then the source will be unloaded from env 1 after it has been moved to env 2 etc. As the migration stops at the Master Environment, the SYSOBJH function will be called twice, once for the Master and once for the From environment.
- **PATH CHOICE –** The user may decide on whether to UPLOAD, DOWNLOAD or UPLOAD ONLY during the initial migration step. PATH CHOICE set to Y will allow the user the choice of an UPLOAD or DOWNLOAD, a PATH CHOICE set to U will force the user into an UPLOAD migration only, with no choice, while PATH CHOICE set to N will default to the normal migration process. i.e. N = during the first migration step the objects linked in the ISR will automatically be downloaded from the MASTER to the INITIAL environment, once the code has been amended in the INITIAL environment, the migration will once again move up the line in an UPLOAD movement. Y = during the first migration step the user will be requested to enter a ‘U’ for Upload or ‘D’ for download. If D is selected, the default process as set up in PATH CHOICE = N will be followed. If U is selected, the objects linked in the ISR will NOT be brought down from the MASTER environment, but will move directly from the INITIAL environment. Please note, that if U is selected, the code must be available in the initial environment for movement, else the ISR will abend. If the object is NEW, it will automatically have a choice of U set, as there will be nothing in the MASTER to download. If a profile needs to be SUSPENDED, then enter a path choice of an ‘S’. This will allow no further use of the profile, but it will be available for history purposes (restores via CC200 or enquiries).
- **IMPORTANT NOTE referring to path choice –** if U is selected, and the object is therefore NOT brought down from the MASTER, there will be no backup of this object in the initial environment and if a restore is needed, it will be restored from the previous backup version of that particular object.

- **COMMENT –** This indicator must be set to **'Y'** or **'N'**. If set to Y a comment will be added to the end of the actual source of a migrated object, referring to the ISR number and date of migration. If set to 'N', a comment will be NOT added. Comment of 'Y' is only allowed for Development Environments and if a compile option is selected, where Development Environments refer to any environment that resides on the True Development Server.
- **MIGRATE USER –** This user will be used in the migration of objects to and from environments, irrespective of who is signed on when the migration takes place. All copies, backups, restores and compiles will be carried out by this userid. The Userid or Group Allowed will continue to be validated even if a Migrate User has been entered. This Userid or Group Allowed is a security based option, ensuring that only specified users are allowed migration. A MIGRATE USER does not prevent a user from migration, but ensures that only the migrate user will be used in the actual transaction of migration. This MIGRATE USER must be a valid user and must be authenticated, else the migration will not be successful. See below for example of authentication. Note, however, that the Userid doing the migration must still be authenticated for remote shell commands, as it will be used in the creating of temporary work files during a migration process.

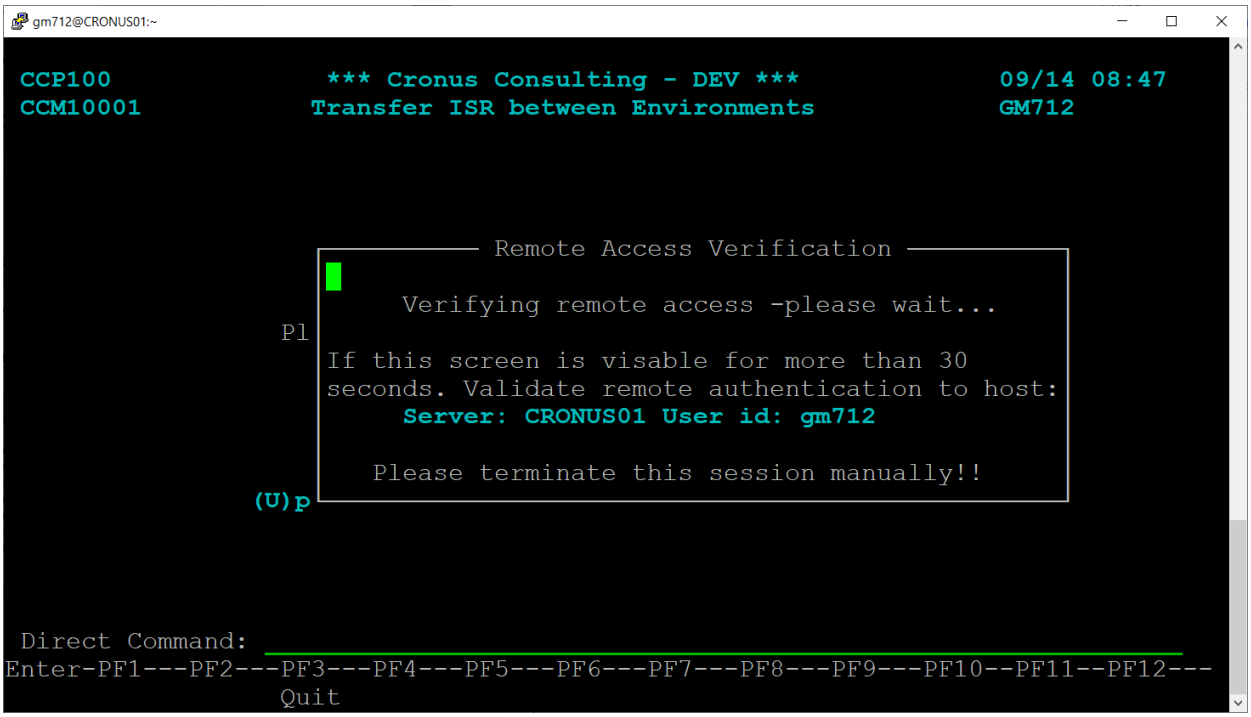
Authentication as follows:

Secure Shell Authentication When using a Migration User-id

- The LUW ssh security definitions located in the **~migration-user-id-home/.ssh/authorized_keys** file, should be duplicated to all users required to execute function CC100.
- The LUW administrator should facilitate this access.
- The administrator can use the following command to validate the access:
  From your user-id on O/S level execute:
  ssh <your-server> -l <migration user-id> ls –l

  Example:
  ssh cronus-dev -l sag ls –l

If the User-id has not been authenticated, then a window will be displayed during a migration using function CC100, and this window will cause the user to "hang" and the migration will not continue. See function CC100 for further explanation. See screen print below. To test this, before taking a migration function, use PF5 in CC001 signed on as the required user.



Authentication Window in CC100

3.1.3 CC010 – ISR Code Maintenance

This function is used to maintain codes that are specific to the run environment of all EspControl processes and should be administered by the person responsible for setting up and maintaining the system run environment.

ISR CODE TYPES:

Code Type	Description
ADHOCL	Default **ADHOC,E,99,99** where 1st code is Natural Library/SCL User to be used for Adhoc moves via CC200 (example ADHOC library). The 2nd code can be **E,Y or N**. If **N** then Adhoc module moves cannot be run via CC200. If **Y** then Adhoc module moves are allowed by any user who has access to CC200. If **E**, then the USERID must be added to RESTOREPRD and only this list of users is able to do Adhoc module moves. 3rd code must be the Production Environment number as seen on CC001 and the 4th code must be the Development environment number as seen on CC001.
APPLIC	SCL Users allowed in EspControl. If only default **SCLUSER** is loaded, then this code is not used. Else load applicable **SCL USER's** which are available for use in EspControl.
APPROVE	ISR Approval users. If only default **APPUSER** is loaded, then this code is not used. Else load **USERID's** who can approve ISR's via CC060. If user is not on this list, cannot execute CC060.
AUDITRPT	**Y** (audit report for CC100 will be sent to the printer) or **N** where no report is required.
CALLBACK	List of Callback users who can run Emergency ISR's. If only default **CALUSER** is loaded, then this code is not used. Else load **USERID's** who are able to run Emergency ISR's.
COMPDEBUG	**Y** will not delete temporary log files during a CC100 migration, **N** all temporary logs will be deleted after each CC100. Not to be set on permanently, only to use if the reason for an error cannot be determined.
COMPLETE	**Y** to automatically complete the ISR after a CC100 move, when entering on the window after a successful move to Production. **N** to allow the user to complete via CC070.
DEFAULTPRF	Set with a default **PROFILE** that will be pulled in for each ISR creation in CC030/CC035, or **NONE** for no default profile. This profile must be an existing profile and will be used for all ISR creations. This can however be overridden in CC030.
EMERGENCY	**Natural Library** to be used for Emergency ISR's and Emergency move via CC200.
ENVIRON	Contains the next number to be used as a pre-numbered Environment and not the environment number on the Control Variable definitions (used in background of EspControl). Cannot be amended.
ENVUSED	Contains a Y or N and if set to Y, Environment will form part of available change control paths. If set to N, the environment is not used and will not form part of available change control paths.
ESPBATCH	Contains a Y or N and if set to Y, SCL transfer will be allowed, if set to N SCL transfer will not be available.
INITBACK	Contains a **Y** or **N** and if set to **Y** the initial backup routine may be run and normal migration may not, and if set to **N** normal migration may continue and the initial backup may not.
ISRNO	Contains the next ISR no to be allocated when an ISR is created via CC030.
ISR'STATUS	Provides a list of possible status's that will be applicable to an ISR.
ISRTYPE	Provides a list of possible ISR types. ISRTYPE is used to group an ISR according to the type of development. NAT, OLD and UBK are system defined types and cannot be amended.
NOSTOW	**Y** allows a profile choice where modules are NOT stowed in Production. **N** forces a profile choice of stow for Production.

PRFCALL	**Prefix** of Emergency names to be used if CALLBACK is used e.g. **EMERG**
RESTOREPRD	List of Users who can restore to Production vi CC200. If only default **PRDUSER** is loaded, then this code is not used. Else load **USERID's** who can select the Production restore via CC200.
RETAPP	**Y** to require Approval via CC061 to return ISR's via CC102. **N** to allow immediate use of CC102 without approval.
RULES	**CALLBACK** = **Y** (Use CALLBACK user list) or **N** (do not use CALLBACK list). **EMERGENCY**= **Y** (Include rules for Emergency moves) or **N** (do not check any Emergency profiles/ISR's for specific rules). **POPCC35** = **Y** (Pre-population of ISR Description and hours in CC035) or **N** (Ignore pre-population. **RESTOREPRD** = **Y** (Use RESTOREPRD user list) or **N** (do not use RESTOREPRD list).
SAMEAPPROV	**Y** for allowing the user who creates the ISR to also approve/migrate the ISR. **N** requires that the creator of the ISR cannot also run and approve it. If set to S and any amendments are made to the ISR, then CC100 checks the date and time of the last approval and if not greater than the save date and time, it requires further approval via CC060.
SAMESTAGE	Contains a **Y** or **N** and if set to **Y**, staging area may be duplicated across environments, and if **N**, then different staging areas must be set up. SAMESTAGE should not be set to Y unless all environments are on different servers.
SCLADHOC	**CODES** – default **E,99,99** where 1st code can be **E,Y** or **N**. If **N** then SCL Adhoc moves cannot be run via CC200. If **Y** then SCL Adhoc moves are allowed by any user who has access to CC200. If **E**, then the **USERID** must be added to SCLADHOC and only this list of users is able to do SCL Adhoc moves. **ADHUSER** is the default template user but not used in CC200. 2nd code must be the **Production Environment** number as seen on CC001 and the 3rd code must be the **Development environment** number as seen on CC001.
SECURITY	Contains an **A, N** or **Y** and refers to the ISR Security indicator in CC030/CC035, if code = **A**, indicator may be marked with **Y** or **N**, if code = **N**, the indicator may ONLY be marked with **N**, and if **Y**, the default will be **N** when creating the ISR, but may be amended by the user to **Y** if necessary.
STORECC	Library names added here will be allowed to be used in the scan function in CC050 option PF2, without doing an actual check to see if the source exists. This will only create the entry in the Inventory List using the current module type for the object that already exists in the Inventory List. These will typically be call-out libraries. CC050 option PF5 will also validate the STORECC list of libraries whereby the module will be scanned in from the actual library but will be created using the STORECC.
SUPERUSR	Enter the **USERID** for the user who in emergency situations may amend ISR profiles that have open ISR's linked. In normal circumstances, only the USERID or GROUP may be altered if open ISR's exist. Enter **NONE** when this function is not required.
SYSTEM	Provides a list of defined systems within the development environment. The SYSTEM variable may also be loaded with "codes" which can be used to control the GENERIC library set up if **GENERIC** is used in any profiles. In this manner a GENERIC library may be set up per ISR as SYSTEM is linked to each ISR. For example, this could be used if multiple Developer libraries are used. See **ISNUX100** for how to set up if SYSTEM is to be used to control GENERIC libraries. System **ACC** is the default system and just refers to Access and will not be used in any generic coding.
VALIDPATH	**Y** will validate the fuser paths/objects at the start of an ISR. **N** will skip the validation for any non-production ISR's. When moving to the Production environment the paths are always validated. This option of N, will speed up the time it takes to migrate an ISR. However, all paths used including the backup/staging areas for all Natural libraries need to have already been created.
XREFNO	Contains a **Y** or **N** and if set to Y will force user to enter a XREF No when creating an ISR, and if N, then XREF is optional. This number is normally a site-specific number

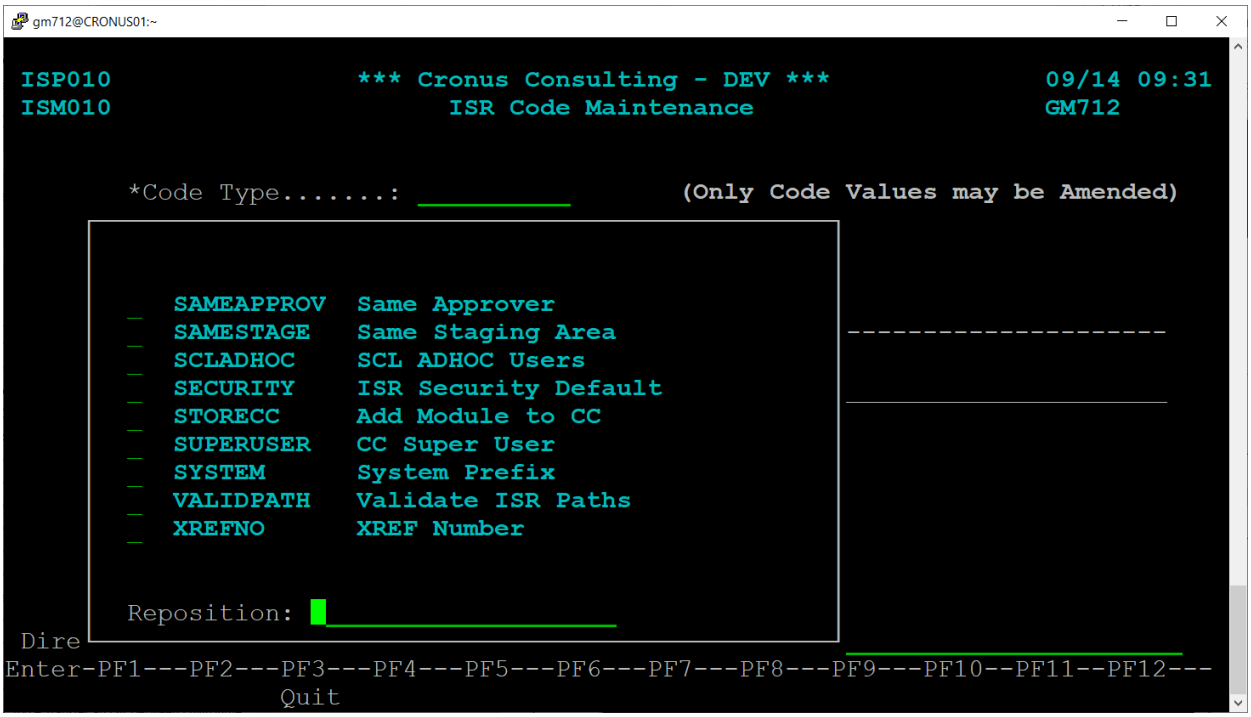
	and there is a user-exit available to validate this number if required, ISNUX030. This routine may be amended per site and exists in library ESPSOFT. Instructions will be found in ISNUX030 itself.

```
gm712@CRONUS01:~

ISP010                  *** Cronus Consulting - DEV ***              09/14 09:31
ISM010                        ISR Code Maintenance                   GM712

        *Code Type.......: __________        (Only Code Values may be Amended)

        _  ADHOCL      ADHOC Library
        _  APPLIC      App SCLUSER security
        _  APPROVE     ISR Approval Users
        _  AUDITRPT    Audit Report
        _  CALLBACK    Callback Users
        _  COMPDEBUG   Compile Debug
        _  COMPLETE    Auto Complete ISR
        _  DEFAULTPRF  Default Profile
        _  EMERGENCY   EMERGENCY Library
        _  ENVIRON     Environment No

        Reposition: ___________________

 Dire
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

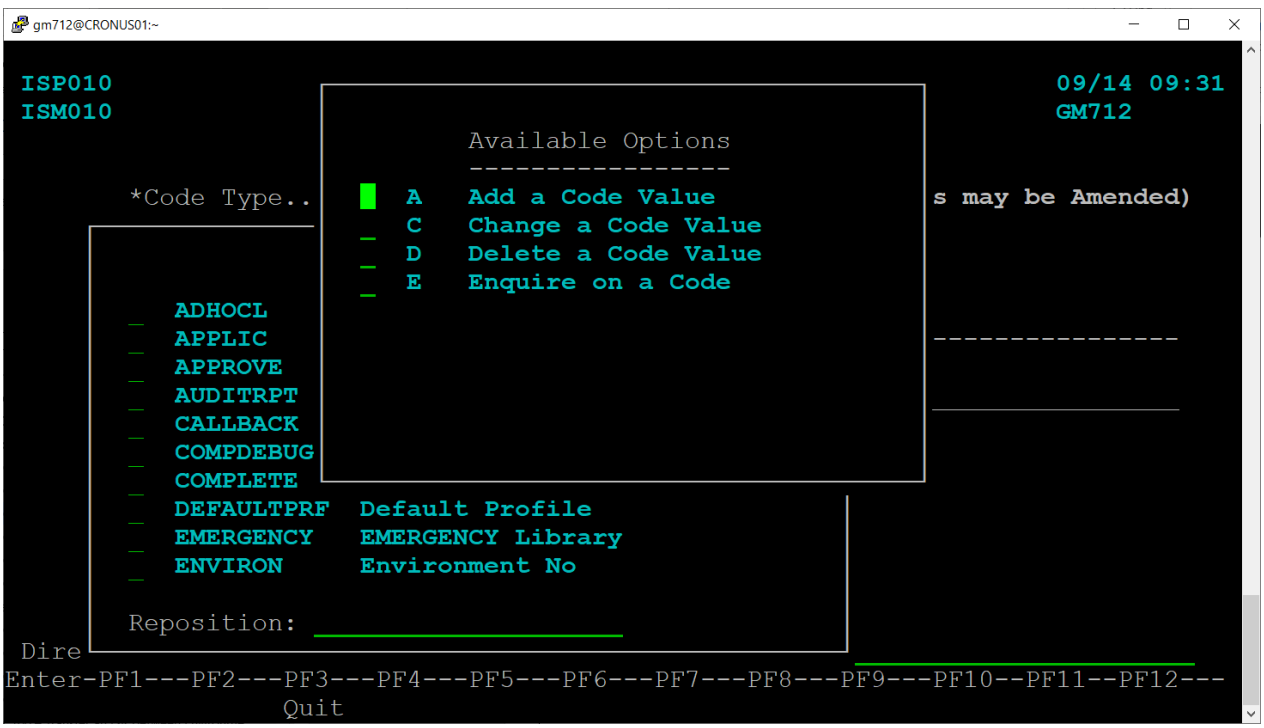
ISR Code Types as specified in table above

```
gm712@CRONUS01:~

ISP010                  *** Cronus Consulting - DEV ***              09/14 09:31
ISM010                        ISR Code Maintenance                   GM712

        *Code Type.......: __________        (Only Code Values may be Amended)

        _  ESPBATCH    EspBatch Running
        _  INITBACK    INITIAL Backup
        _  ISRNO       New ISR Number
        _  ISRSTATUS   ISR Status
        _  ISRTYPE     ISR Type
        _  NOSTOW      No Stow in Prod
        _  PRFCALL     Callout Profiles
        _  RESTOREPRD  Prod Restore Users
        _  RETAPP      ISR Return Approval
        _  RULES       RULES

        Reposition: ___________________

 Dire
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

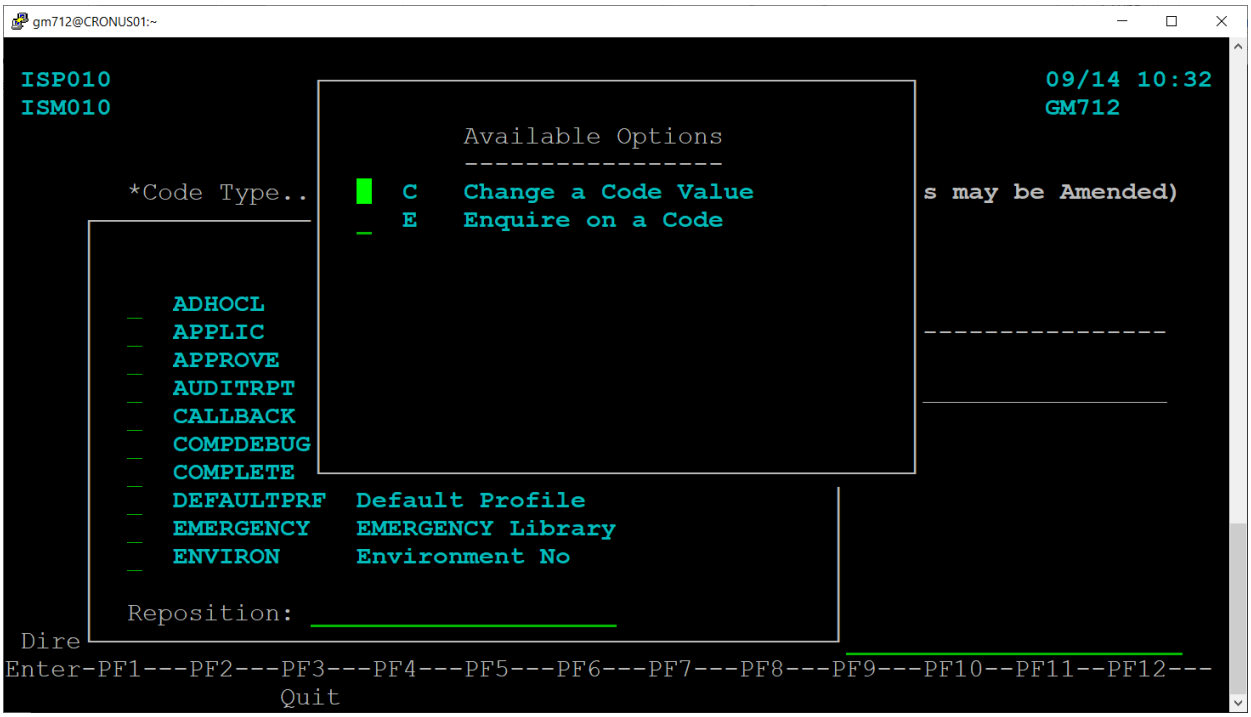
More ISR Code Types as specified in table above



More ISR Code Types as specified in table above



Available options for Selected Code Type for e.g. CALLBACK



Example of a different subset of options for a specific Code Type for e.g. COMPDEBUG

ISR CODES are predefined and may not be added to or deleted, only CODE VALUES associated with these code types may be amended and their functional options are determined by what type of ISR Code is being selected. Therefore, only specific codes may be amended or code values added to. By selecting the required code, the help function will assist the user in reflecting what options are allowed, either Add, Change, Enquire or Delete. See two examples above of different code types having different option selections.

Example 1: CALLBACK – new users may be added or deleted for this code, so this ISR code may have many code values, in this case “userids”.

Example 2: COMPDEBUG – code value may only be amended to either a Y or N, no additions allowed.

Special Notes:

- COMPDEBUG – when any migration (transfer, return or restore) is executed, the objects get compiled, stow’ed or saved, depending on the profile indicators. These compile-type routines generate scripts across servers. If this indicator is marked as ‘N’, these scripts will not be removed at the end of a migration. If marked with ‘Y’, these scripts will remain for debugging purposes. It is a good idea to leave this indicator as an ‘N’, until a problem occurs that requires compile debugging. This can then be turned on, and the ISR rerun.

- AUDITRPT – if set to Y, after each successful migration, irrespective of the environment from or to, an audit report (resembling all details on CC100 window) will be automatically sent to the print

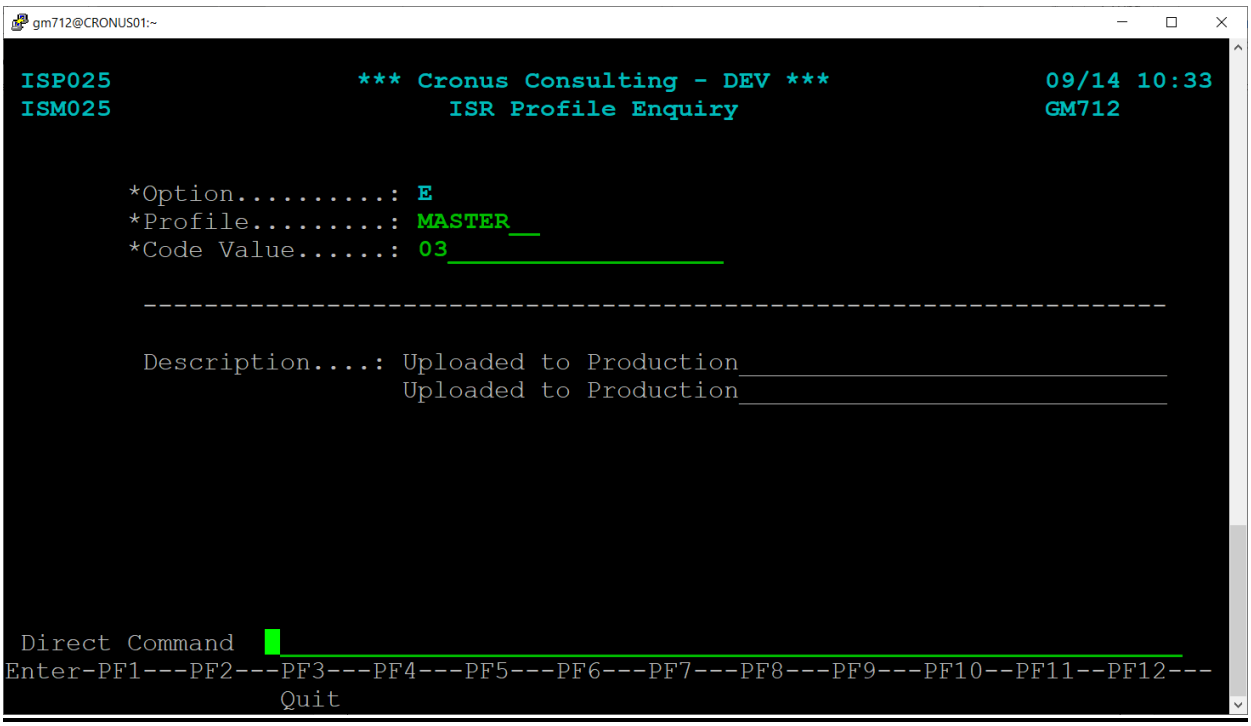
queue. This will detail all successes, as well as all errors and error codes. See CC100 for sample screens.

- ISRTYPE – In order to archive an object, the ISRTYPE must be selected as OLD. Use function CC250 and not CC050 and CC100 with ISR's that have the ISR'STATUS set to OLD. If an ISR is created with type OLD, normal migration will not be possible via CC100 and an error will be returned. In the same manner CC250 will return an error if the type is NOT specified as OLD.
  In order to run an initial backup, the ISRTYPE must be selected as UBK, this will be run via CC350 and as explained above, an error will be returned if CC100 or CC250 is run with this type and vice versa.
  If SCL transfer is available, the ISRTYPE must be selected as SCL. This will differentiate between normal ISR's and SCL ISR's and will execute the correct routine from each function selected.

- ESPBATCH – during installation select "Y" if EspBatch is running in environment or 'N' if not. If ESPBATCH = 'Y', SCL selection, copy, transfer, profiles, return and display functions will be available.

3.1.4 CC025 – ISR Profile Enquiry

This function is used to enquire on Profiles and depending on the profile path, the status codes will be reflected. It must be noted that each ISR may contain a different path, therefore more than one ISR may contain the same status, but this status will reflect a different environment. All enquiries show the status dynamically with the correct environment path per code. The function is for clarification purposes.



ISR Profile Enquiry

Therefore, status 03 as in the example above means Uploaded to Production, but another profile that only has 2 environments in its path will reflect status 02 as being Uploaded to Production. When checking an ISR status, always refer to the profile to get full clarity. The status will refer to the environment number as selected in a specific profile, and not the environment number set out on the Master in CC001.

3.1.5 CC300 – Scan Program Source-code

This function is used to populate the EspControl object inventory and update the “new-object” indicator to “old” on all the objects being scanned in. The function should **only** be executed during the installation procedure of EspControl – or when duplicate copies are made of production libraries on development and the user wants the object to be scanned in as OLD. All objects scanned will be recorded as existing objects in all the environments, which means that the ISR transfer flow will **start** from the Master Index environment to the Initial Environment specified on the profile. If more than one Development environment is specified, this is controlled by the environment number being scanned in from (as reflected below). This means that an object does not have to exist in ALL Development Environments, but at least one, and will be controlled by the profile.

```
gm712@CRONUS01:~
Program source-code scanned !!!
 CCP300              *** Cronus Consulting - DEV ***              10:36:01.1
 CCM300        Scan Program Source-code (Initial Scan ONLY)       2022/09/14
                                                                  GM712




                  Enter library name..: YOURLIB_
                  Start program.......: DEMO0000
                  End program.........: DEMOZZZZ
                  Scan environment....: 01 - Development


     Enter * in library name to scan ALL libraries in chosen environment




 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Initial Source Scan

If the Master source-code environment is on a remote system you would have to set the DBID on the EspControl DDMs to reference the Development ESP Database before you execute CC300 – that is, if Entire Net-Work is available – if not **Please contact us to discuss the best options to do the initial source code scan.**

If SCL migration is available, SCL’s are not scanned in via CC300, but via a new function CC330. See Appendix A function CC330 at the end of the manual, for detail explanation of how to scan in SCL’s.

All types of Objects (e.g. Programs, Maps, Text etc.) may be scanned and will be updated in the Object Inventory with the correct Object Type as they exist in the fuser. A range of programs may be entered and all objects in between, will be scanned in, for example DEMO0000 thru DEMOZZZZ. If the entire library is required enter only * and * in both the start and end program parameters.

If a new object is developed and needs to be part of the Inventory List, this CC300 function must NOT be used, but the new object must be scanned in via PF6, function CC050 as explained below in the manual.

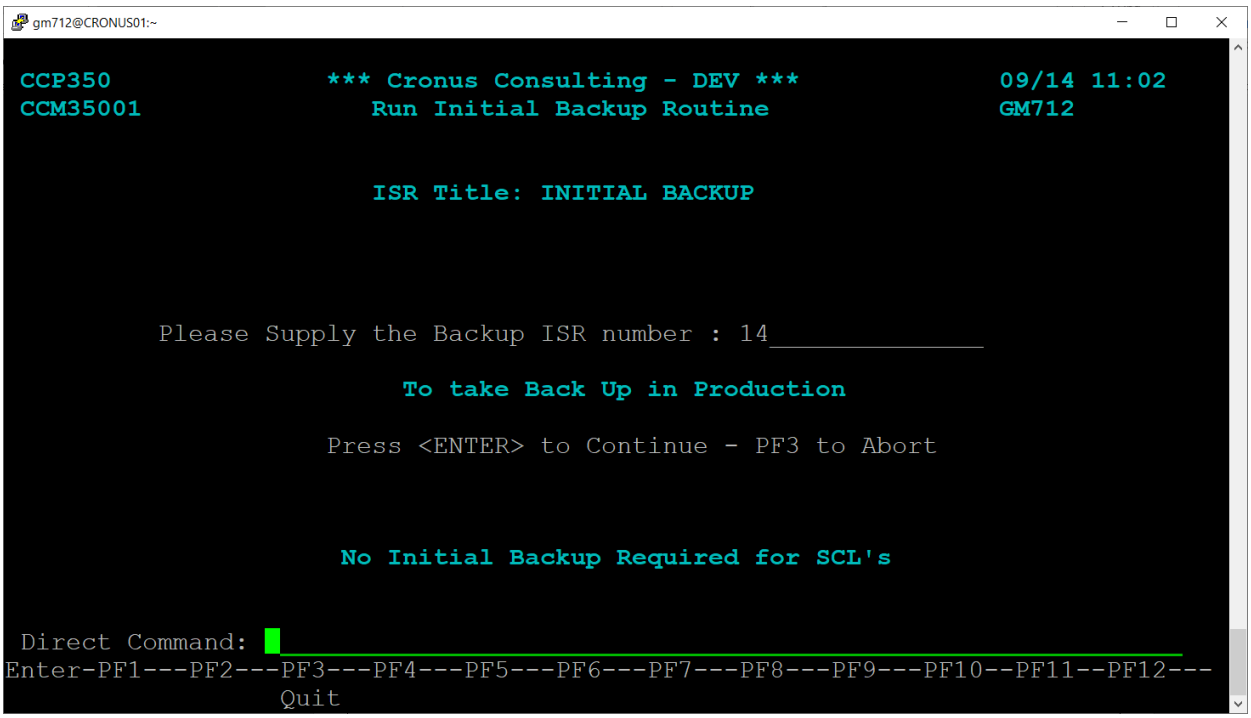
3.1.6 CC330 – Scan SCL's for Transfer

This function is used to scan in SCL's in the same manner as objects are scanned in via CC300. See Appendix A for full explanation of CC330.

3.1.7 CC350 – Run Initial Backup Routine

This function is used to run an initial backup of all objects in a choice of environments, so that a formal EspControl Backup will exist before a new install or an upgrade. As the backup cycle (see CC100 as explained below in the manual), takes the backup when moving FROM an environment, if a problem occurs the very first time an object is migrated, there will be no initial backup. This routine will create such a backup. **Please note, an Initial Backup is highly recommended, as if not run and an ISR aborts which contains an object which is being moved for the first time, no restore will be done and a manual restore will have to be done. This could cause a problem if the ISR aborts while being migrated to the production environment or the user ran a migration on an ISR that has a profile of UPLOAD.**

CC350 runs in the same manner as CC100, it must be executed once for each environment needed to be backed up. While CC350 is in the process of running or has not yet completed the backup for all selected environments, the normal migration of objects via CC100 will NOT be allowed and CC100 will issue an error message if run.

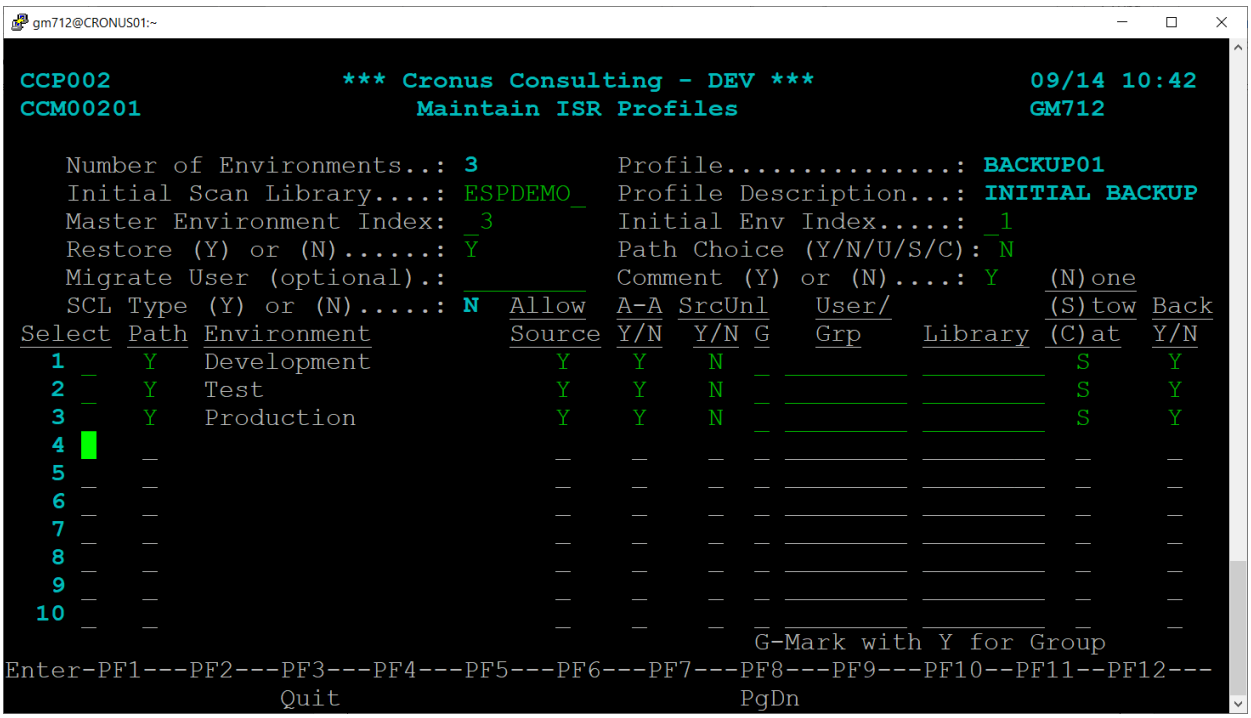


Run Initial Backup Routine

The initial backup routine has specific steps that must be completed in a specific order, before the above CC350 may commence running. Some of these steps will include transactions explained further in the manual.

STEP 1:

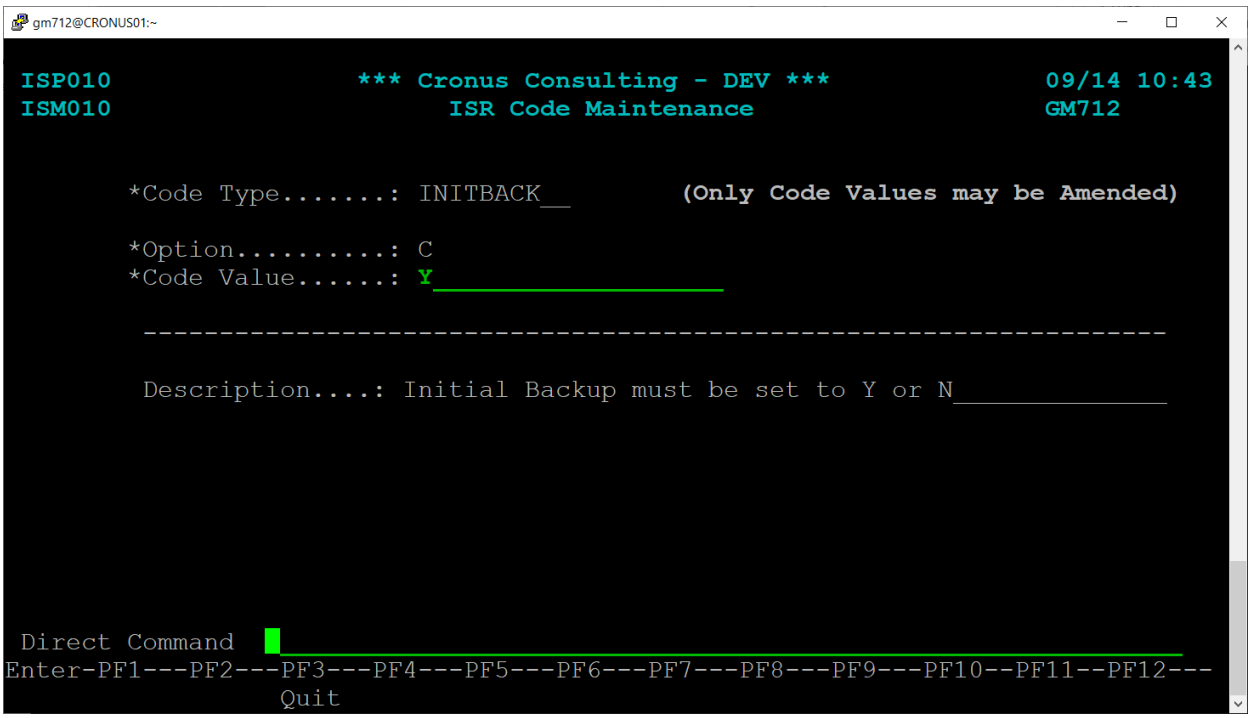
- CC002 - Create a profile, ensuring that you have ‘BACK’ as part of your profile name e.g. BACK01, BACK02 etc
- This profile must have all paths linked that you want to have backed up. E.g. if you want Dev, Test and Prod to be backed up, then PATH = Y for all of the environments. If only Dev and Test, then only mark PATH = Y for these two.
- The profile must have ALLOW-SOURCE = Y and BACKUP = Y. All other indicators may remain as the default as they are ignored.
- Ensure that the ISR-LIBRARY is correctly set up, as this library will determine what library is to be backed up, if specified. If not specified, the library that the object was scanned in with will be used for all environments being backed up. The OBJECT LIBRARY is not used in the initial backup and cannot be set via CC050.
- GENERIC ISR Libraries may also be selected (see CC002 for GENERIC explanation)
- With a BACK* profile, the library for the initial environment (i.e. True Development) may be amended (this is disallowed with all other profiles). By amending the ISR LIBRARY in Backup profile, a different backup may be run for each library in each environment (a different ISR number), if required.



CC002 Step 1: Create BACKUP profile

STEP 2:

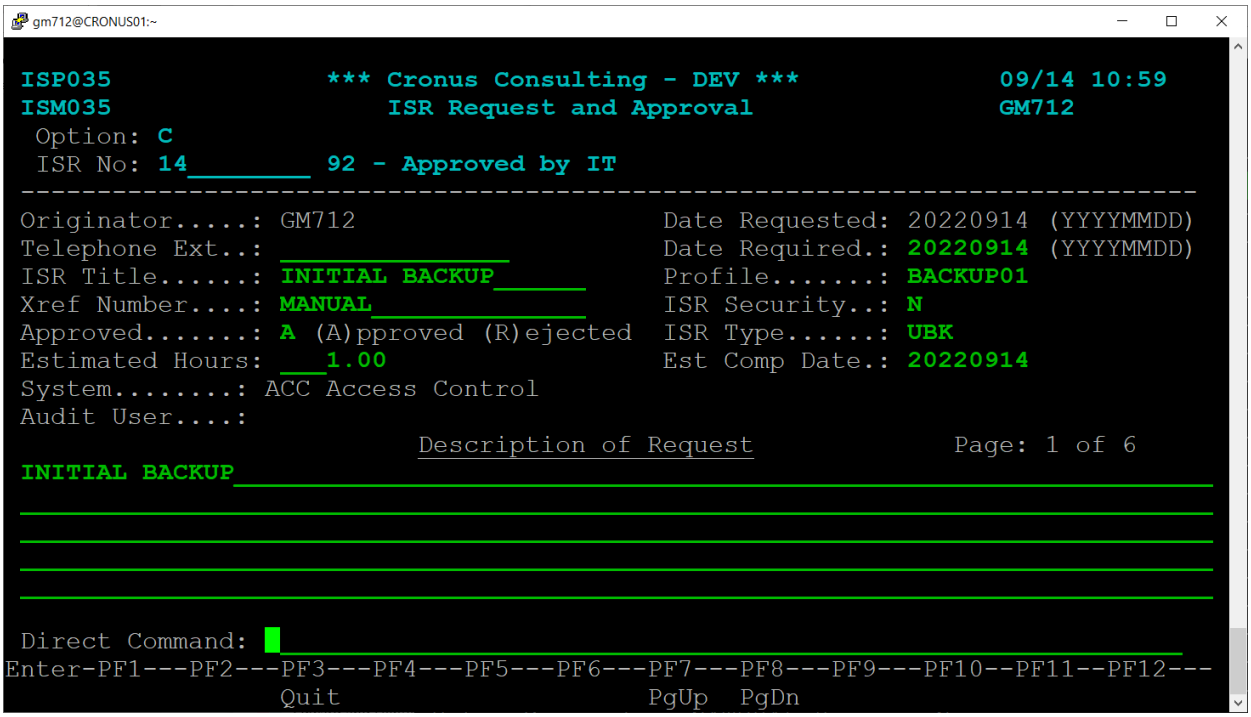
- CC010 – Check ISR-CODE INITBACK and it will be set to Y after an install, but will still be N if an upgrade has been done (Y is the default from the install program that is run initially). If you do not want to run an initial backup, then set this to N and EspControl will not allow you to run CC350. If it is set to N and you want to run an initial backup after an upgrade, set to Y. Once CC350 is complete, EspControl will automatically set INITBACK to N. If you wish to run another initial backup, you can manually amend the code to Y and redo CC350.



CC010 Step 2: Set INITBACK code

STEP 3:

- CC035 - Create an ISR with automatic approval and choose ISR type UBK as your ISR type. Link your “BACK01” profile to this ISR.
- The ISR will automatically link the objects when running via CC350 and will run as a normal ISR does – it creates history, displays objects etc. and completes as a normal ISR, after running once for each environment specified in the BACKUP profile, but only does a backup of the module, no actual copying occurs.

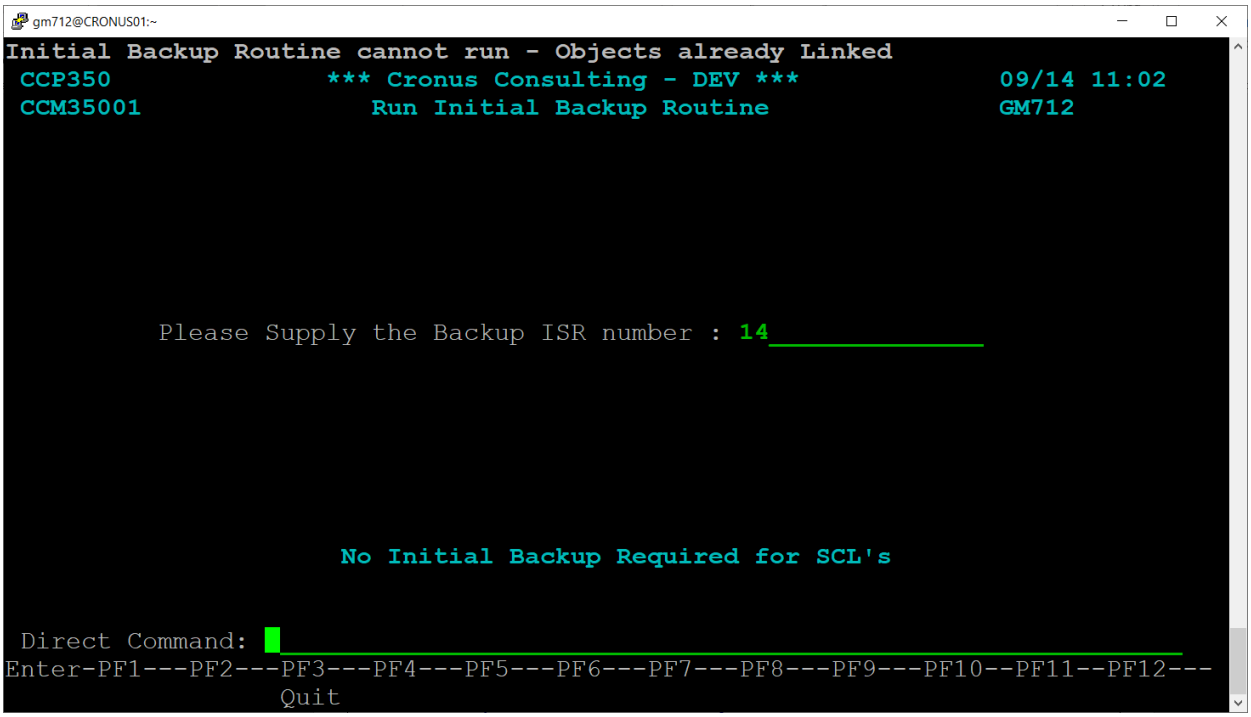


CC035 Step 3: Create and Approve the ISR

STEP 4:

- CC350 – Run the initial backup function, various validation checks will be done and only if all these checks are clean will the routine commence.
- Run CC350 for as many times as number of environments linked in the backup profile. CC350 will execute one backup per environment at a time.
- Only UBK ISR’s created and linked to BACK* profiles will be allowed. The backup routine MAY NEVER be run when any objects are linked to other ISR’s. It must run as a stand-alone routine, with all objects being open for backup. This will also be validated.
- ISR Code INITBACK (CC010) must be a Y.
- No Objects are linked physically in CC350, as in a normal ISR (See CC050 as explained below in the manual). The CC350 routine does this for you, depending on the ISR library captured in the BACK* profiles.

- The objects being backed up will reflect on the screen and also the status of the backup. Y for backup correctly and N for no backup if an error has occurred. CC350 will never halt if the objects do not exist or could not be backed up correctly. A report will be sent to the spooler at the end of the routine for any objects that were not correctly backed up. These may also be viewed in the history transaction CC095 (explained below in the manual).
- CC350 will also do a path validation of the backup paths used in the running of CC350. If these backup paths do not yet exist, they will be automatically.



CC350 Step 4: Validation check for Linked Objects - Error

```
gm712@CRONUS01:~
CCP350                    Run Initial Backup Routine                 09/14 11:05
CCM35002                   Taking Backup From Test                   GM712
  To        ISR NO: 14 - INITIAL BACKUP
Library   Object     Backup    Back Lib                                    Resp
ESPDEMO   TST3             Y   ESPDEMO
ESPDEMO   ZGL1             Y   ESPDEMO
ESPDEMO
ESPDEMO
ESPDEMO                        Backup Taken in Test
ESPDEMO
ESPDEMO                       Press <Enter> to Continue
ESPDEMO
ESPDEMO
ESPDEMO   ZGL4             Y   ESPDEMO
ESPDEMO   ZGL5             Y   ESPDEMO
ESPDEMO   ZGL6             Y   ESPDEMO
ESPDEMO   ZGL7             Y   ESPDEMO
ESPDEMO   ZGL8             Y   ESPDEMO
ESPDEMO   ZGL9             Y   ESPDEMO
ESPDEMO   ZGLERR5          Y   ESPDEMO
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC350 Step 4: Detail screen reflecting objects being backup up

STEP 5:

- CC070 - Once CC350 has been run for all linked environments, the ISR will complete via CC070 in the same manner as a normal migration routine (see CC070 as explained below in the manual).
- All the objects will be un-linked, the completion flag for the ISR set on and the INITBACK code set to N. Continue with EspControl.
- These backups will then be used as a “previous backup”, if your ISR abends on a specific object and it is the first time the object is being migrated.
- CC200 can be used to restore from the initial backups as well, if required.

NOTE: New Objects are not included in the Initial Backup. At install time all objects would already be existing objects and if the initial backup is run again at upgrade time or whenever requested, NEW OBJECTS do not need a backup as they only exist in Development and will create the first backup once being moved from Development. The New Object Indicator seen in CC050, as explained below in the manual, will determine this.

```
gm712@CRONUS01:~
CCP350                    Run Initial Backup Routine                 09/14 11:02
CCM35002                Taking Backup From Production                GM712
  To        ISR NO: 14 - INITIAL BACKUP
Library   Object     Backup    Back Lib                                      Resp
ESPDEMO   TST3            N   ESPDEMO
ESPDEMO   ZGL1            Y   ESPDEMO
ESPDEMO
ESPDEMO
ESPDEMO                 Backup Taken in Production
ESPDEMO
ESPDEMO        Backups Did Not Complete - SEE REPORT for list
ESPDEMO                  Press <Enter> to Continue
ESPDEMO
ESPDEMO   ZGL4            Y   ESPDEMO
ESPDEMO   ZGL5            Y   ESPDEMO
ESPDEMO   ZGL6            Y   ESPDEMO
ESPDEMO   ZGL7            Y   ESPDEMO
ESPDEMO   ZGL8            Y   ESPDEMO
ESPDEMO   ZGL9            Y   ESPDEMO
ESPDEMO   ZGLERR5         Y   ESPDEMO
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC350 Step 5: Completion screen for each environment

```
gm712@CRONUS01:~
<ENTER> to complete ISR
ISP070                *** Cronus Consulting - DEV ***                 09/14 11:06
ISM07001                Mark an ISR as Completed                      GM712

   ISR Number: 14
   ISR Status: 99 - Completed
-------------------------------------------------------------------------------
   Profile:     BACKUP01    INITIAL BACKUP
-------------------------------------------------------------------------------
 ** If Manual Complete and objects should not be restored, Mark with N **
            Restore.........: N (Y/N/P - Yes/No/Previous ISR)
-------------------------------------------------------------------------------

           Actual Work Hours:   1.00

            Completion Date.: 20220914

Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC070 Step 5: Final completion after all linked environments have been backed up

```
gm712@CRONUS01:~
Job number 1108598: N.20220914.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/14          List of Objects with INCOMPLETE Initial BACKUPS -          >
11:03:59.8                     http://www.cronus.co.za

Backup Environment : Production

          Object        Env          Object        Env
           Name         No            Name         No
     ----------------  ---      ----------------  ---

     ESPDEMO TST3        3      EMERGENCZGL1        3
     EMERGENCNOMAP       3      EMERGENCZGL4        3
     EMERGENCZGL2        3      EMERGENCZGL14       3
     ESPDEMO DEMO01      3


   These Objects were not backed up in the Backup Routine (CC350)

   Determine why no backup was taken, if necessary take manual backup OR
   Create another UBK ISR and rerun entire CC350, amend ISR Code INITBACK to 'Y>
```

Initial Backup Error Report

3.2 ISR Transfer Function Overview (CC501)

3.2.1 CC030 – ISR Request Capture

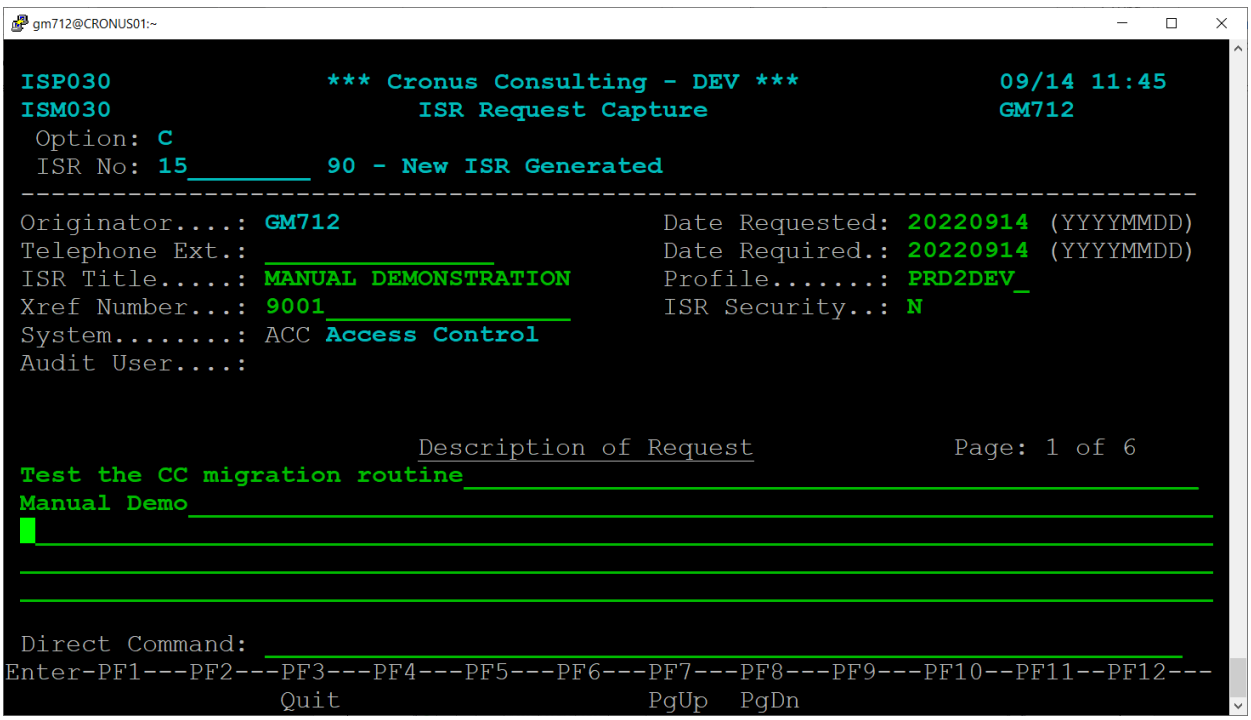
This function is used to capture and maintain an ISR (Information Service Request). The ISR detail should be completed either by the business analyst responsible for the request or by the developer on behalf of the originator of the service request.

Function Options:

A - Add ISR

C - Change ISR

E - Enquire on a ISR



ISR Capture

When adding a new ISR (Option **A**) the system will automatically generate the next ISR number using the ISRNO Code Type found in CC010. This number is used internally by EspControl and referenced throughout the application. If the Comment Indicator is switched on, then EspControl automatically inserts a reference at the end of the source object during the migration between environments. The ISR Number, ISR Title and Dev Team Member (from CC040) fields are inserted as the comment line. Option "**C**" is used to maintain the ISR detail. Access to the ISR detail is restricted as soon as the ISR is approved.

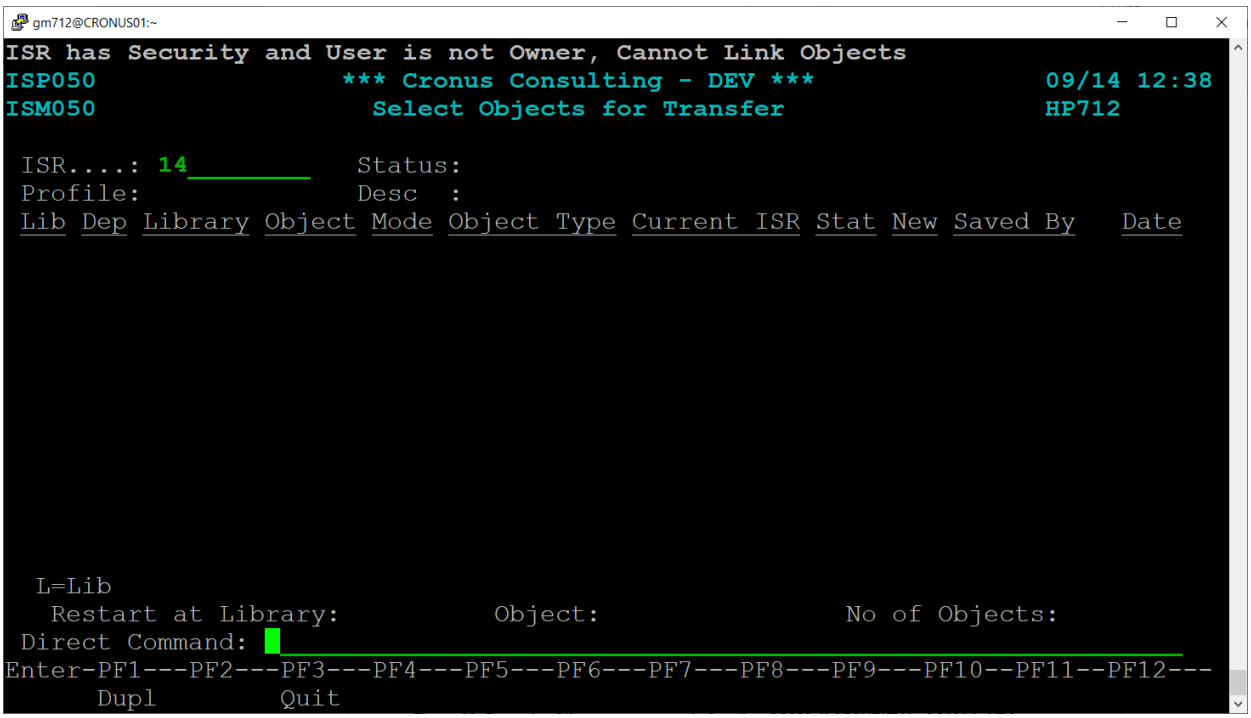
The following information must be entered for an ISR:

- Originator - Set to the User-ID of the person capturing the ISR
- Date Requested - The date that the ISR was requested.

- Telephone Ext. - The telephone number or extension of the person that initiated the ISR. Optional.
- Date Required - The date when the ISR should be completed.
- ISR Title - A brief description of the ISR.
- Profile - This must be a valid Profile (enter for help on Profile names).
- Description - A detailed description of all requirements pertaining to the request. The description is a scrollable area – PF7 and PF8 can be used to page backwards or forwards.
- XREF Number - The Project Number or Ticket Number used by the client to link the ISR to their own Project Control system. If the ISR CODE XREFNO is set on, then this x-ref is mandatory. This number can be verified by the client in coding inserted in the user exit ISNUX030, and then if not valid an error will be returned. If this is not required, leave ISNUX030 as it is.
- ISR Security - Must be Y or N. If “Y”, then security is switched on and only the originator may approve (CC040), link objects (CC050) and transfer objects (CC100). If set to N, no user-id restrictions, but menu restrictions will be used. The ISR Code SECURITY can be set to determine defaults (see CC010 as explained above in manual).
- System - Enter for help on System Code, enter system affected by ISR. The System may also be used to assist with the creation of GENERIC libraries (see CC010 for code SYSTEM and ISNUX100 for updating of these GENERIC libraries). In this manner a different GENERIC library may be set up for each ISR.
- Audit User - Audit user will be automatically updated by the user creating the ISR.

The ISR will be migrated using the path and all the data captured on the Profile (see CC002) linked to this ISR. Therefore, every ISR set-up could be linked to a different profile and will follow its own path using its own set of indicators to determine Source, Restore, Backup, Stow/Cat and User-id functionality. The ISR No will be used through the entire EspControl system, linking objects, restore capabilities, history enquiries, status summaries and migration routines together.

Only ISR Status – 90, 91, 92 (created, approved and new) will allow specific detail to be amended, once the ISR is in migration status, only certain detail may be amended, depending on the status. The long description, XREF number and ISR Security may still be amended via CC030, once a migration move has taken place and the status does not conform to the ones specified above, but ONLY by the originator of the ISR or the approver of CC040. Any other user will be disallowed from amending these. Use CC040 to amend the long description if required by the approver. Once the ISR has been completed, NO detail amendments will be allowed by any user, originator or not.



Example of the error message in function CC050, if SECURITY in CC030 is YES

3.2.2 CC035 – ISR Request and Approval

This function is used to capture and maintain an ISR (Information Service Request) and do the approval at the same time. It is a combination of both CC030 and CC040. The process, rules and validation of this function will be the same as explained in CC030 and CC040. If CC035 is used to create the ISR then there is no need for approval via CC040. The only time CC040 will be used, is to link an SCL ISR to the created ISR. This is explained in CC040 below.

Function Options:

A - Add ISR

C - Change ISR

E - Enquire on a ISR

```
gm712@CRONUS01:~
ISR security must be (Y)es or (N)o
 ISP035              *** Cronus Consulting - DEV ***            09/29 06:16
 ISM035                  ISR Request and Approval               GM712
  Option: C
  ISR No: 33________ 90 - New ISR Generated
 ---------------------------------------------------------------------------
 Originator.....: GM712                    Date Requested: 20220929 (YYYYMMDD)
 Telephone Ext..: ______________           Date Required.: 20220929 (YYYYMMDD)
 ISR Title......: NEW ISR COMBINE_____     Profile........: PRD2DEV_
 Xref Number....: APPROVAL___________      ISR Security..: N
 Approved.......: A (A)pproved (R)ejected  ISR Type......: NAT
 Estimated Hours: ___1.00                  Est Comp Date.: 20220929
 System........: ACC Access Control
 Audit User....:
                          Description of Request            Page: 1 of 6
 NEW ISR COMBINATION TEST WHEN CREATING.______________________________________
 ______________________________________________________________________________
 ______________________________________________________________________________
 ______________________________________________________________________________
 ______________________________________________________________________________

 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit                    PgUp  PgDn
```

ISR Capture and Approval

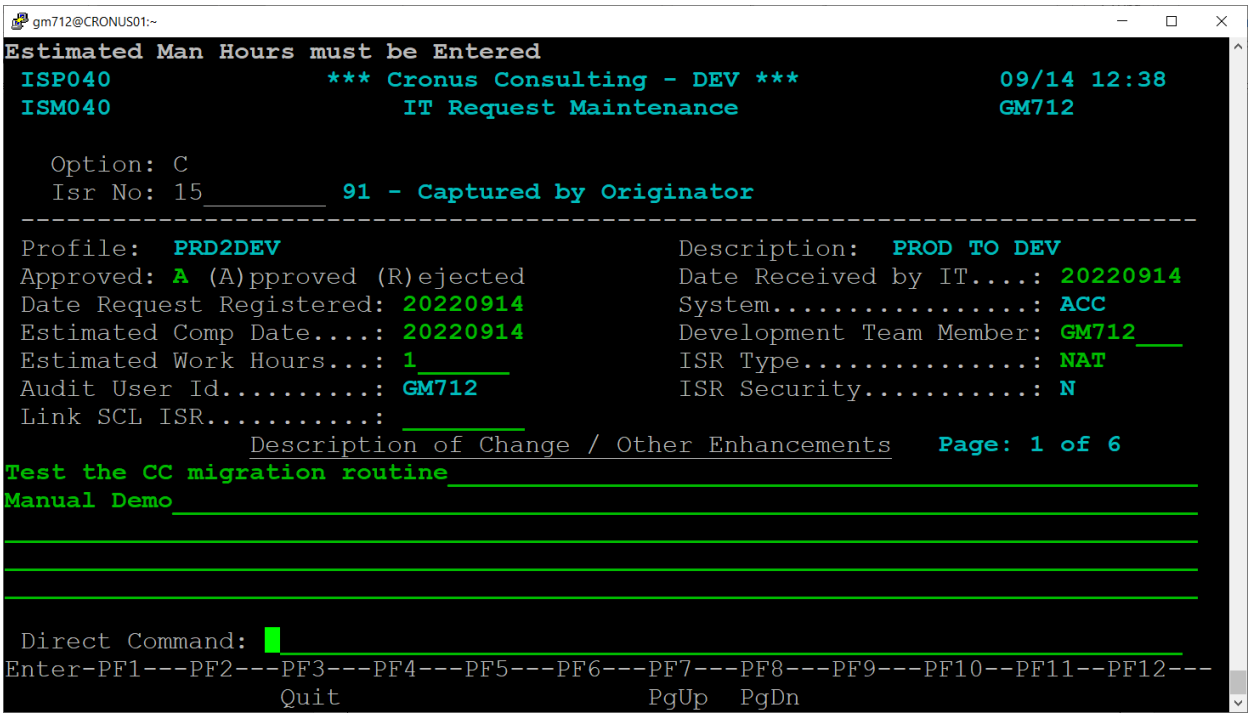
3.2.3 CC040 – Approve ISR Request

This function is used to capture all IT related information that pertains to an ISR. This function is also used to approve or reject an ISR before any migration occurs. If rejected, the ISR will not be allowed for any further linking of objects or migrating of objects. It will be in the EspControl system for History purposes only as a request not allowed.

Function Options:

C - **Change an ISR** or

E - **Enquire on an ISR**



ISR Approval

If the ISR SECURITY field has been marked with a “Y” in CC030, then only the user who created the ISR in CC030 may approve this request.

- **The following data, that is not populated from CC030, must be entered for an ISR:**
- Approved - Approve / Reject the ISR (Y or N)
- Date Received by IT - Set to System Date
- Date Request Registered - Date that ISR was requested
- Estimated Comp Date - Date by which request will be completed
- Development Team Member - Developer assigned to the request
- Estimated Manhours - Time required completing the ISR

- ISR Type - Type of request – selected from the code ISRTYPE
  Always select OLD for Archiving.
  Always select UBK for Initial Backup.
- Description - May be added to or amended here, once blocking of CC030 detail has occurred

3.2.4 CC050 – Link Objects to an ISR

The function is used to select all objects that form part of the requested ISR.

```
gm712@CRONUS01:~

ISP050                 *** Cronus Consulting - DEV ***                09/14 12:42
ISM050                   Select Objects for Transfer                  GM712

 ISR....: 15_________     Status: 00 - Object Selection in Progress
 Profile: PRD2DEV         Desc  : PROD TO DEV
 Lib Dep Library Object Mode Object Type Current ISR Stat New Saved By   Date
X _   _ ESPDEMO  DEMO01   R  Program            15  99      GM712    20220914
_ _   _ ESPDEMO  DEMO02   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO03   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO04   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO05   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO07   R  Program                        Y  GM712    20220914
X _   _ ESPDEMO  DEMO08   R  Program            15   1   Y  GM712    20220914
X _   _ ESPDEMO  DEMO09   R  Program            15   1   Y  GM712    20220914
_ _   _ ESPDEMO  DEMO10   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO11   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO12   R  Program                        Y  GM712    20220914
_ _   _ ESPDEMO  DEMO13   R  Program                        Y  GM712    20220914
  L=Lib                     ** Start of Data **
   Restart at Library: _______ Object: ________       No of Objects: 3
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Dupl  AddCC QUIT  Error ScanD Scan        PgDn        Curr
```

Object Selection for Transfer

Once the ISR No has been entered a list of all objects will be displayed that can be linked to the ISR. If an object is currently linked to another ISR it will still be displayed but will **not** be available for selection.

Objects can now be linked by entering a 'X' next to the object in the New Column. The number of objects that are currently linked to the ISR will be updated at the bottom right of the screen during object selection. By marking the **Dependents** column for a given object all dependents for the selected object will be marked automatically (EG. If the Object contains a FETCH and a CALLNAT, both objects will be selected and form part of the ISR). However, if these dependents are already linked to another ISR, they will be ignored.

Please note: Dependents are not available for LDAs, PDAs and GDAs. The Dependents function is only available if SOURCE exists in the scanned Development environment and has not been moved out via Allow-source = N or = M. In this case, the user must mark the dependents manually to become part of an ISR.

PF10 acts as a toggle key to display all objects or just the objects linked to the current ISR.

Once an object has been selected the Current ISR column will be updated to reflect that the object is now linked to the current ISR. The Status displayed under the Status column will indicate in which environment the object currently is. The status of an object will constantly be updated as transfers take place across the defined environments. Depending on the profile per ISR, each status could reflect a different environment.

For example, 01 could reflect DEVELOPMENT on one ISR and Test on another ISR if the second ISR was linked to a short-path profile (from Test to PROD only).

When an object is selected the status will be updated to one of the following values:

- 99 - The object is currently in the master environment and will be downloaded to Initial Environment (Existing Object) or Upload only as specified by the profile
- 01 - New Object (Does not require download from master environment as new objects are created in development)

If the ISR status is greater than 01, it indicates that the object was previously selected and the ISR has already been transferred. This does not prohibit the linking of additional objects to the ISR.

If an object is deselected from the ISR, a choice is given to select a restore of the object or not. If NO RESTORE is selected (mark with N), the object is unlinked, the object status is reset, but the object itself will remain where it is in the migration path of the chosen ISR. e.g. if the object had been moved from Dev to Test and then deselected with no restore, the object will remain as is, in both Dev and Test. If RESTORE is selected (mark with Y for Restore from current backup version or P to restore from Previous backup version), the object is unlinked, the object status is reset and the object is restored using the backup version selected above to all the environments processed in the migration path of the chosen ISR. For Restore = Y, the backup version used will be the one from the current ISR or if no backup exists, the previous version backed up in the previous ISR. For Restore = P, the previous version will always be used irrespective if a current backup exists or not. Once an object has been removed from the ISR it will be available for selection by other ISR's. If the indicator ALLOW-SOURCE has been set to 'N', the object (compiled) code will be restored to all affected environments, but the source will only be restored to the initial environment of the profile linked to the ISR. This will allow the user to work on the object source again before being re-linked to another ISR. Source is always restored to the Staging Area in all environments, irrespective of the ALLOW-SOURCE indicator. If the indicator ALLOW-SOURCE has been set to 'M', then the source and object will be restored back to the environments. The restoring window will reflect as in function CC110 in this document.

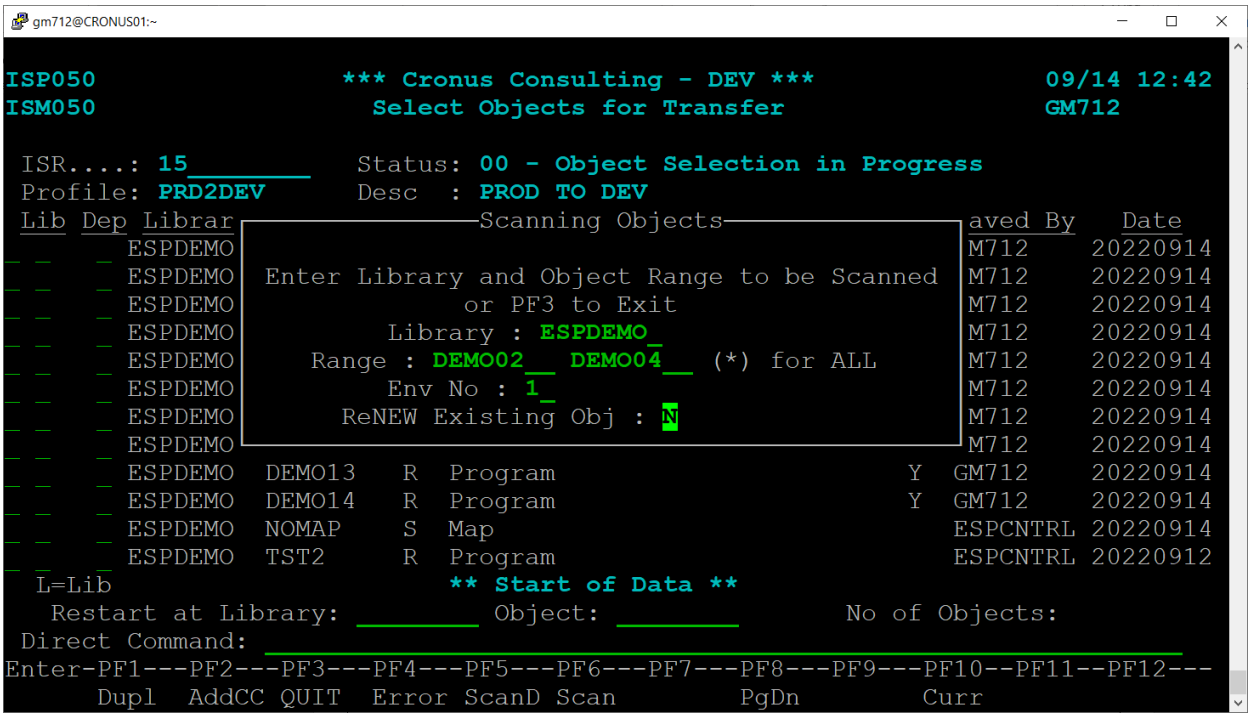
```
gm712@CRONUS01:~
ISP050                 *** Cronus Consulting - DEV ***                09/14 13:18
ISM050                   Select Objects for Transfer                  GM712

 ISR....: 15_________    Status: 01 - Downloaded to Development
 Profile: PRD2DEV        Desc  : PROD TO DEV
 Lib Dep Librar                                                  aved By   Date
  _   _  ESPDEMO                                                 SPCNTRL 20220914
 _ _   _ ESPDEMO                LIBRARY: ESPDEMO                 M712    20220914
 _ _   _ ESPDEMO                OBJECT.: DEMO01                  M712    20220914
 _ _   _ ESPDEMO                                                 M712    20220914
 _ _   _ ESPDEMO  ARE YOU SURE YOU WANT TO REMOVE THIS OBJECT    M712    20220914
 _ _   _ ESPDEMO                                                 M712    20220914
X _   _  ESPDEMO   Enter (Y)ES or (N)O      : Y                  M712    20220914
X _   _  ESPDEMO                                                 M712    20220914
 _ _   _ ESPDEMO   Restore unlinked Object : Y                   M712    20220914
 _ _   _ ESPDEMO                                                 M712    20220914
 _ _   _ ESPDEMO    (Y)ES OR (N)O OR (P)rev                      M712    20220914
 _ _   _ ESPDEMO                                                 M712    20220914
  L=Lib
   Restart at Library: _________ Object: _________        No of Objects: 3
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Dupl  AddCC Quit  Error ScanD Scan        PgDn        Curr
```

Unlinking of Objects with Restore options

The scan function (**PF6 Scan**) is used to scan NEW objects in from a specific Natural library and will add this object to the EspControl inventory, to be available for further selection. The function key **PF6** is used to initiate the scan of source code library and scan environment specified, and to update the EspControl inventory with objects added. When scanning via PF6, the object must exist in the environment requested, so that the object type and mode may be picked up. If the object does not exist, it will NOT e added to the Inventory List.

Please note that the **initial scan (CC300)** is still required to build inventory (only for OLD and existing Objects) and using the PF6 option will add or update objects. If an object changes its mode, or object type etc, for example from REPORT to STRUCT, then PF6 must be used to update the mode in the Inventory List. Failure to do this, will result in the ISR aborting as the compile will not work. Using PF6 on an old object will not affect the NEW OBJECT indicator, using PF6 on a new object will update the new object indicator with Y. However, using CC300 on a NEW object WILL affect the new object indicator and the object will be updated into the inventory list incorrectly. If this has been done in error, use PF6 and mark the RENEW indicator with a Y. This will again update the object to NEW.

PF6 may be used with a range of objects using “*” as the range indicator, but if the Renew Indicator is marked with a Y, no range is allowed and the start and end object must be the same. If the START and END object are the same, only the START object needs to be entered.



PF6 – Scan Source Code for New Objects

PF6

Enter the starting from and ending at range **or** * to scan the complete library. Scanning a **complete** library might take **some time,** depending on the number of new objects in the Natural Library – a selection can be entered for a specific range of objects. The library name is validated, but the objects are not, the scanning of these objects will add them to the Object Inventory if they exist or ignore them if they do not. The “Env No” that must be entered is defaulted to TRUE DEVELOPMENT, but may be amended by the user if more than one development environment exists. If an environment is chosen that does not exist on the development machine or does not match the Library entered, the scanning will not work! The Library that is scanned in for each of the development environments becomes the default library of that object. These objects will all be scanned in and the Object Inventory updated with the NEW OBJECT indicator set to Yes if the object is new. If an old object is re-scanned it will re-update the Object Type and Mode, but will not set the New Object indicator on.

NOTE – If the Allow-source = M option is set on Development and an object is scanned for example, to change the object type or mode and the object does not exist, then no updating will take place. This will also be the case for PF1, the duplicating of objects, see section below. Therefore, these objects MUST be present in this environment when they are being scanned or duplicated.

```
gm712@CRONUS01:~

ISP050                 *** Cronus Consulting - DEV ***                 09/19 08:35
ISM050                   Select Objects for Transfer                   GM712

 ISR....: 17_________    Status: 01 - Downloaded to Development
 Profile: PRD2DEV        Desc  : PROD TO DEV
 Lib Dep Librar ----------------Scanning Objects----------------  aved By   Date
X _   _ ESPDEMO                                                   M712     20220914
 _   _  ESPDEMO  Enter Library and Object Range to be Scanned     M712     20220914
 _   _  ESPDEMO              or PF3 to Exit                       M712     20220914
        ESPDEMO          Library : ESPDEMO_                       SPCNTRL  20220914
        ESPDEMO      Range : NOMAP___ ________ (*) for ALL        M712     20220914
 _   _  ESPDEMO          Env No : 1_                              M712     20220914
 _   _  ESPDEMO       ReNEW Existing Obj : Y                      M712     20220914
 _   _  ESPDEMO                                                   M712     20220914
 _   _  ESPDEMO  DEMO13    R  Program                        Y    GM712    20220914
 _   _  ESPDEMO  DEMO14    R  Program                        Y    GM712    20220914
 _   _  ESPDEMO  NOMAP     S  Map                                 ESPCNTRL 20220914
 _   _  ESPDEMO  TST2      R  Program                             ESPCNTRL 20220912
  L=Lib                        ** Start of Data **
   Restart at Library: ________ Object: ________        No of Objects: 1
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Dupl  AddCC QUIT  Error ScanD Scan        PgDn        Curr
```

PF6 – ReNEW Existing Object Indicator set to Y

Changing Library Names

```
gm712@CRONUS01:~

ISP050                 *** Cronus Consulting - DEV ***                 09/19 08:35
ISM050                   Select Objects for Transfer                   GM712

 ISR....: 17_________    Status: 01 - Downloaded to Development
 Profile: PRD2DEV        Desc  : PROD TO DEV

  Display Library      Display Object     To Library     Environment
    ESPDEMO              DEMO04           ESPDEMO_       Development
                                          TEST____       Test
                                          ESPDEMO_       Production
                                          ________
                                          ________
                                          ________
                                          ________
                                          ________
                                          ________
                                          ________

 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Retur                   PgUp  PgDn
```

Library Selection per Object

By marking the **LIB** column with an '**L**', a window will be reflected (see above) and if required to be different from the default, another migration library may be entered. The library from the "TRUE DEVELOPMENT" Environment(s) may **never** change from what was scanned in via **PF6 or the initial setup scan CC300**.

This True Development is environment 01 in CC001 and not the first environment, in for instance, a short-path profile. If the same object-name exists in different libraries in Development, it may be scanned in again via PF6, so that two entries exist in the Inventory. If the client uses different development libraries for each developer, this may be overcome, by creating a DUMMY first environment on CC001 that is never marked as a Y in any profile path. Then scan the objects in using the Production library and use GENERIC in the profile to “change” the library name for each developer as required.

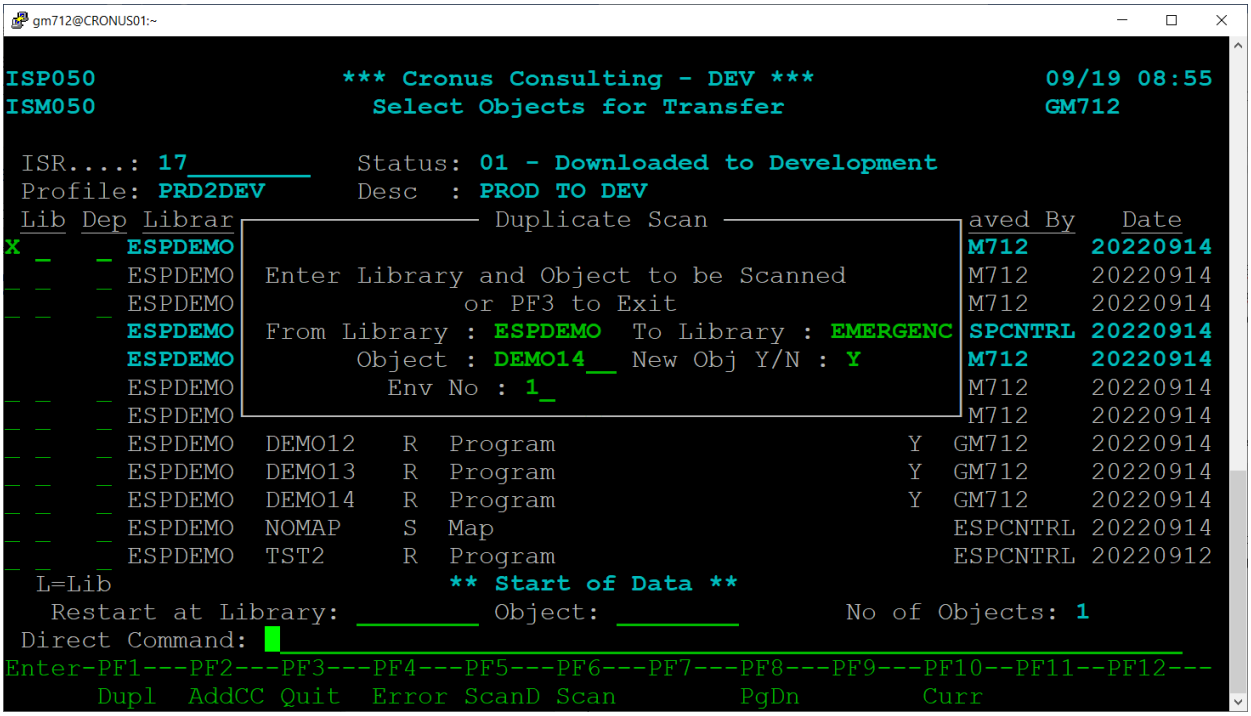
However, migration from the “True Development” library to any **other** library in the objects migration path **is** possible. This library list will be cleared once the ISR is complete and the next time an object is linked to an ISR, a new set of libraries may be selected. The path of the selected objects via the profile and selected libraries linked to these environments will be stored in History and may be viewed on function **CC095** or function **CC088**. If ‘L’ is entered, the library names (if already there or if an ISR Library has been set up in the profile), are displayed, but may be amended accordingly.

This is optional and if no libraries are selected the scanned library will be used for an entire migration path, with the exception of when an ISR Library has been set up in the profile linked to the specific ISR. Therefore, the migration path of an object, with regards to the linked libraries will be as follows:

1. No external libraries added, no ISR Library set up in profile linked to ISR, use OBJECT LIBRARY in Inventory List (scanned) across all environments.
2. No external libraries added, but GENERIC ISR Library exists, use GENERIC user-exit ISNUX100 to determine library.
3. No external libraries added, but ISR Library exists (not GENERIC), use ISR library then Object Library from Inventory List
4. External Libraries added, override both ISR Library and Object Library
5. Restores – use the library that was migrated at the time of the ISR (history kept of this)

The above Library path is always followed during Migration, Return or Restore.

Duplicating Objects



PF1 – Duplicate Scan

This function (**PF1 Duplicate Scan**) is used to create a copy of a selected object, from the EspControl inventory in another library in any Development environment and again update the EspControl inventory with the new object name and library. This new object may now be linked to an ISR and be migrated accordingly.

The function key **PF1** is used to initiate the scan, and in turn will copy the source code of the entered **Object** and **From Library,** to the **To Library**. PF1 only copies SOURCE and the object must be manually compiled by the user.

Object Status

The **New Obj** can be marked as a "Y" or "N" depending on the user requirements. If the original copied object is a new object, then this will be overridden and treated as a new object as well.

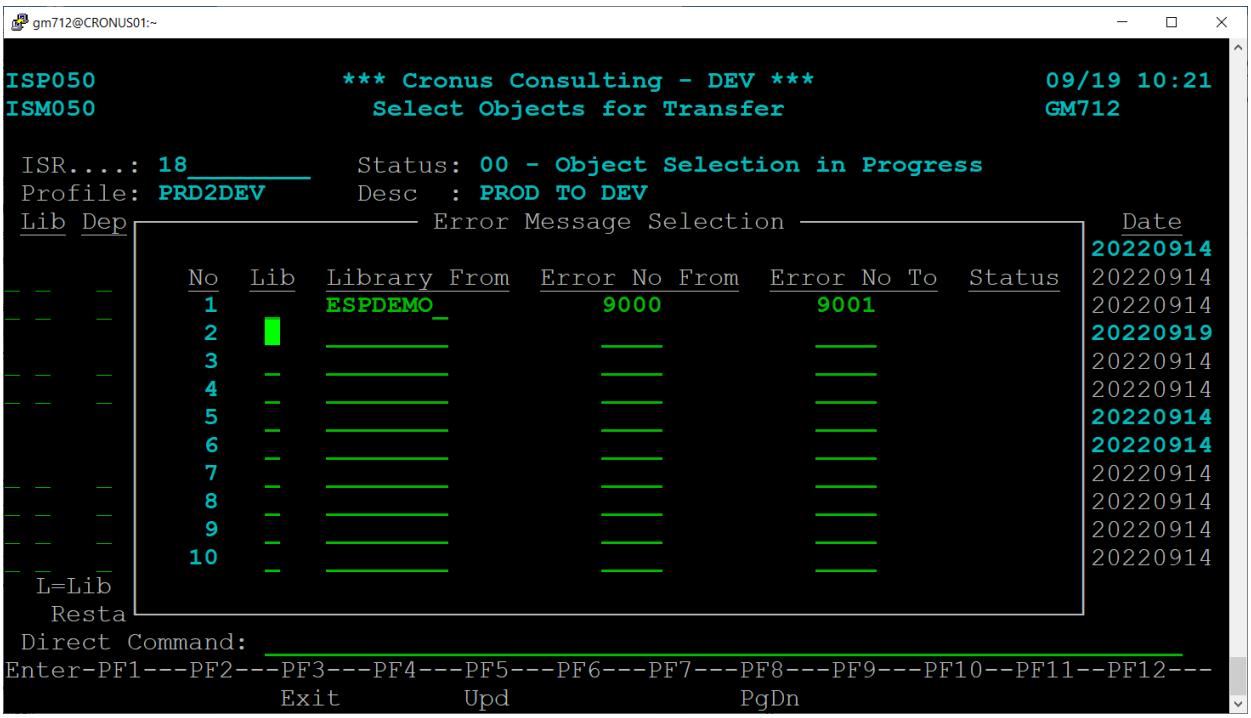
"Y" - Once selected will follow profile path and libraries selected or allocated in column 'L', moving up from Initial Environment to the Master Environment.

"N" – Copied object will be created as OLD, amend Libraries in column 'L' if necessary. However, when linked to an ISR, the migration will attempt to move this object down from Production first. If this is not required, use a Profile with a Path Choice.

In this manner, ISR's may run in tandem with an object, as long as one ISR is one environment ahead of the other.

If an ISR is selected for migration in CC100 whereby an object is going to over-write the same object from another ISR, in the same environment, a WARNING will be given. Use <PF4> to override or <PF3> to reject. This will only occur if objects with the same name, but different Development libraries, are linked to multiple ISR's.

PF4: Setting up Migration of System Error Messages



F4 – Setting Up Error Messages

System Error Messages

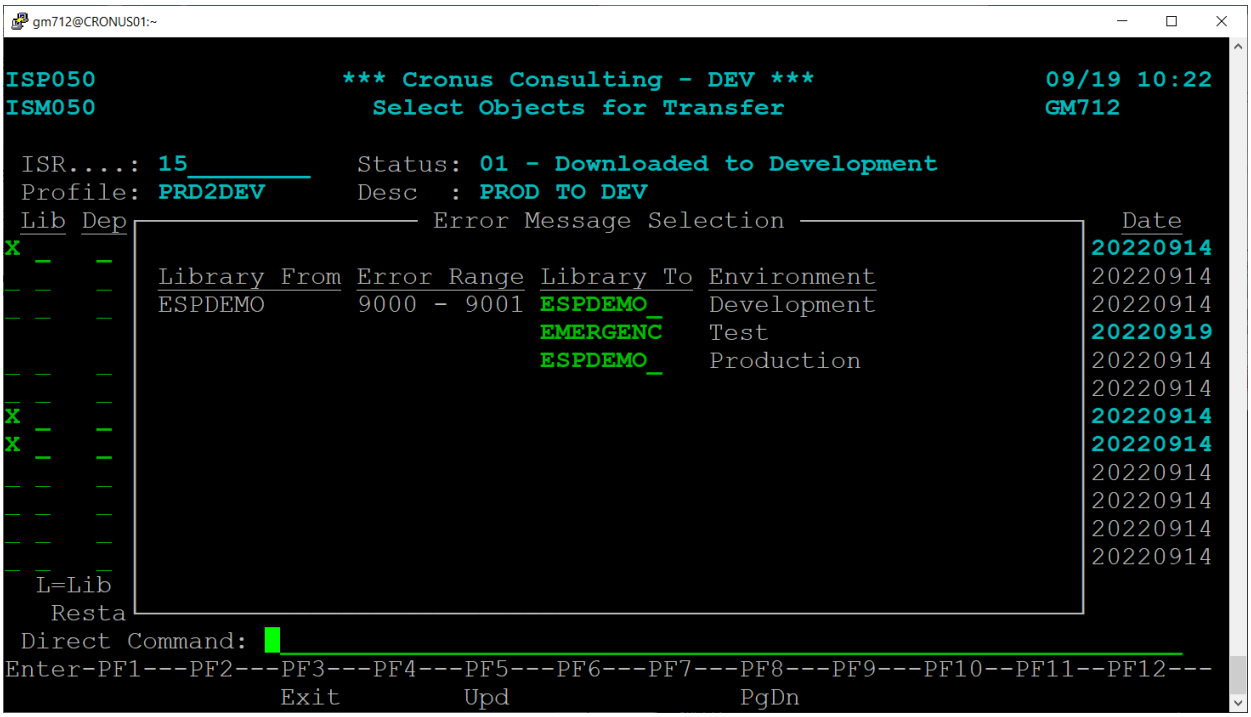
User defined SYSERR messages, may be migrated along with the objects in an ISR. Error messages can be added to the ISR using PF4 in CC050. A window will be displayed as reflected in the figure above. If no objects are currently being linked, set up a dummy program in the environments where the system error messages are being migrated to, for e.g. SYSERRP and then this program may always be used to transfer error messages without impacting on other objects and ISR's.

Multiple libraries may be specified, with varying range of error numbers. These error numbers should be reflected as a FROM or TO number. If only one error is required, then the FROM and TO error numbers will be the same.

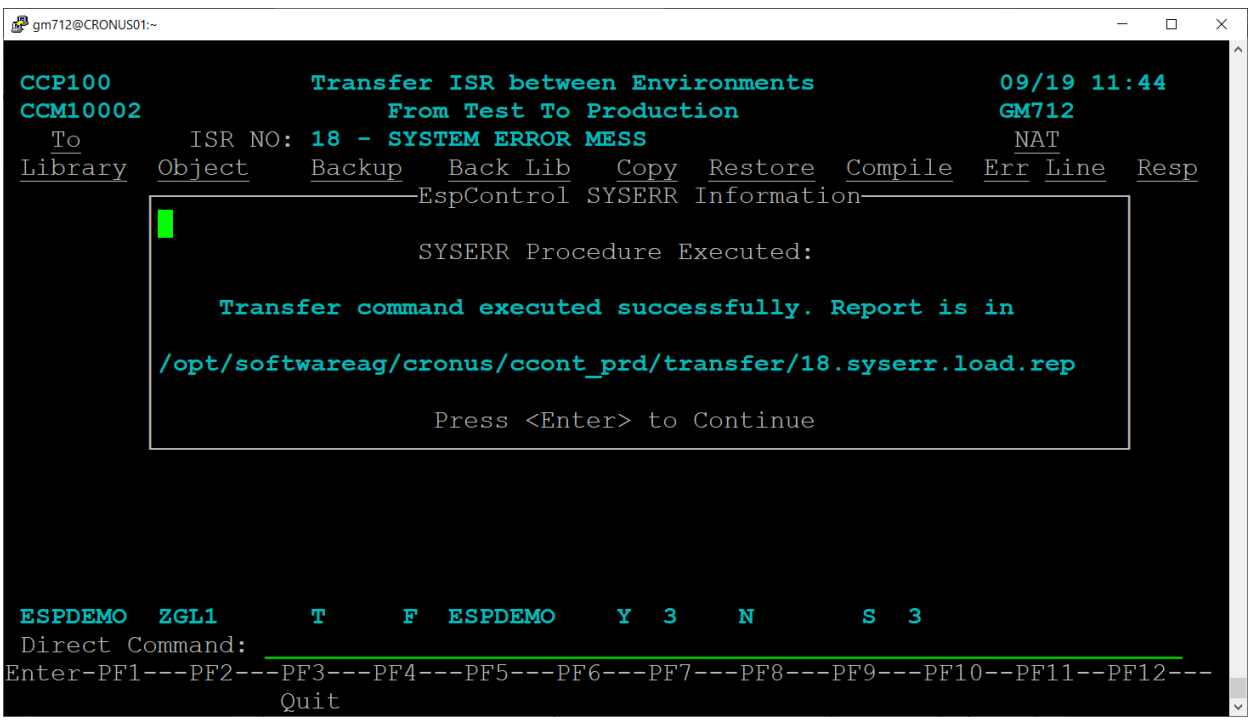
The Libraries that the error messages are being transferred to are specified per ENVIRONMENT.

LIB:	If Libraries that error messages are being migrated to, do not differ from the LIBRARY FROM, then this field need not be used, as the LIBRARY TO will be automatically populated with the same Library as the LIBRARY FROM. If the libraries are to differ, user must enter 'L' in this column.
LIBRARY FROM:	Library System Error Messages are being moved from. This must be a valid library in the Initial Environment.
ERROR NO FROM:	The Starting error number in the range specified.
ERROR NO TO:	The Ending error number in the range specified. The TO ERROR NO cannot be less than the FROM ERROR NO
LIBRARY TO:	The TO LIBRARY specified must be a valid library in the migrated to environment and ALL environments must have a specified LIBRARY TO, as the system error messages will be migrated along with the objects in the same PATH as the linked objects to the specified ISR.

If the linked objects are OLD objects the original error messages will be migrated from the MASTER Environment to the Initial Environment at the start of the ISR. These may then be amended in the Initial Environment and may be transferred again accordingly. Alternatively, only add the system error messages via PF4 once all objects are in the Initial Environment (Status 01).
To save the captured Error Numbers, press 'Y' for SAVE or N for Quit without save. Once saved, these error messages will be migrated when function CC100 is executed for the specified ISR to which this Error Selection has been linked. PF5 will also save changes.



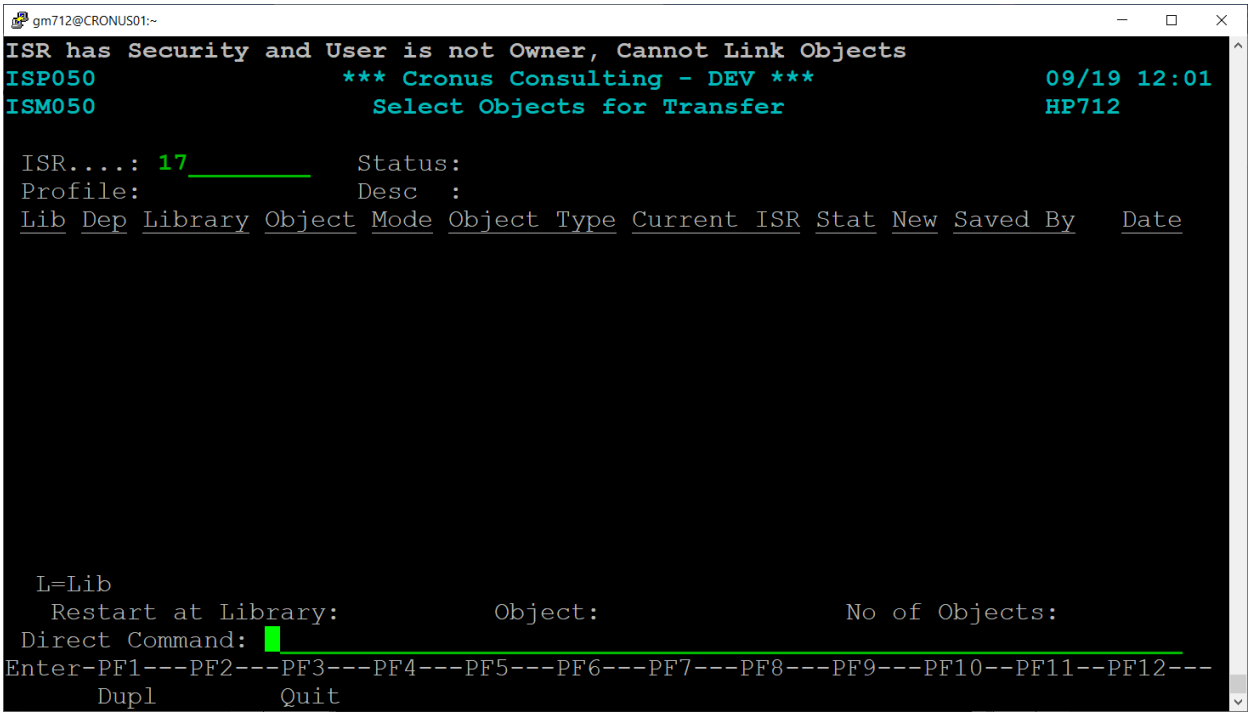
'L' in Lib column, specify Environment Libraries for SYSERR



SYSERR confirmation

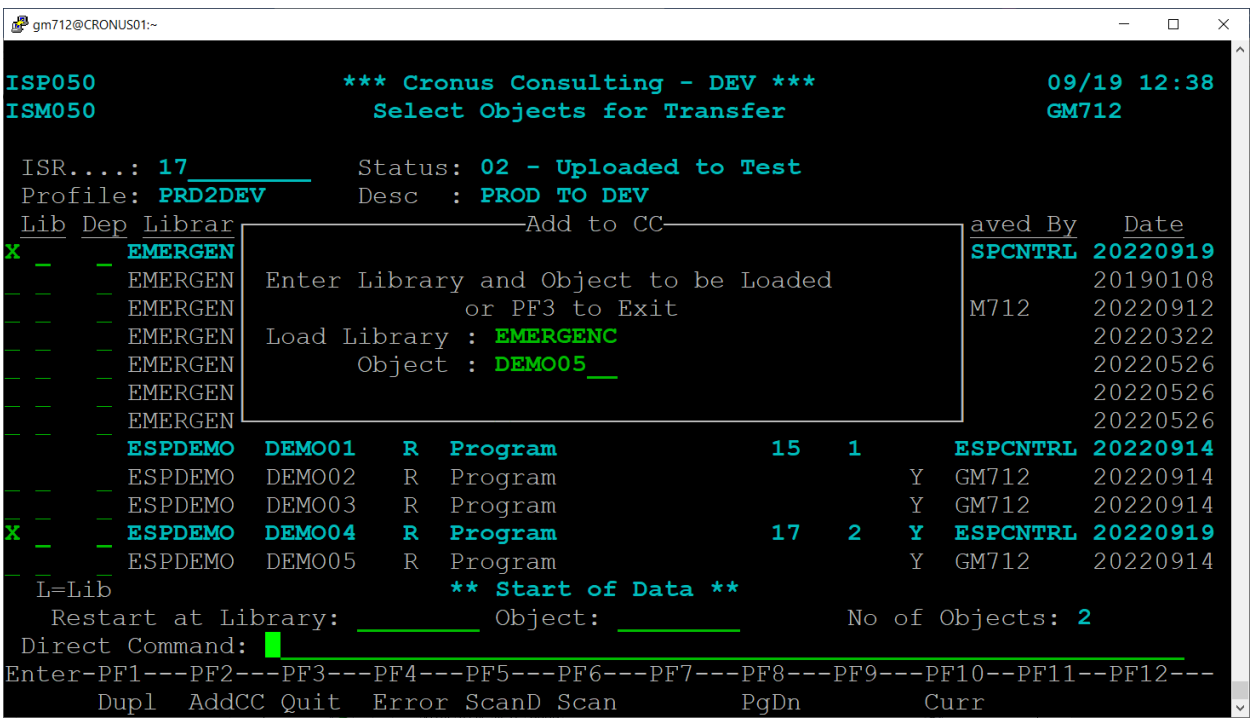
ISR SECURITY – If the ISR Security of the current ISR has been marked with Y, then CC050 may only be used by the same user who created the ISR. This is irrespective of the function security and whether you have the CC050 function allocated to you or not. Even if it is allocated, if ISR Security is on, then any

user other than the creator will receive the error (as specified in figure below) that they are not the owner and cannot continue with CC050.



Error if ISR has Security and user for CC050 is not Owner

Add Emergency Modules to EspControl



PF2 – Add modules to EspControl

This function (**PF2 Add CC**) is used to add modules to the EspControl inventory in CC050, to be used in a migration move, but with a different library name, in Emergency or Call-Out situations. This function will load the module with the input library name and input object name (as seen above) to the Inventory List for linking and migrating. This function would be used in the case of a call-out where the developer is not allowed in the actual Production Library – he will add the module name to EspControl, using the call-out library and then move the code from the actual Production Library to the call-out library, work on the module and then return it to the actual Production Library. In this manner modules added via PF2 could only move in a Production to Production profile from one library to the other.

The validation rules for PF2 are as follows:

- The library entered is validated against the code STORECC in CC010, where all libraries to be used in this function are specified. Failure to load the library name against the STORECC code will return an error when using PF2.
- The module entered must already exist in the Inventory List on CC050 and the object mode and type will be picked up from this already existing entry.
- Once a module has been added to the EspControl inventory list to a call-out library, if used in a call-out situation again, it need not be re-added as the entry will remain for further use.
- The module loaded via PF2 is not scanned and checked for valid source, but the object information is obtained from the current entry in the Inventory List.

Add Emergency Modules to EspControl

```
gm712@CRONUS01:~

ISP050                 *** Cronus Consulting - DEV ***              09/19 12:40
ISM050                   Select Objects for Transfer                GM712

 ISR....: 17_________    Status: 02 - Uploaded to Test
 Profile: PRD2DEV        Desc  : PROD TO DEV
 Lib Dep Librar                 Scanning DevLibs                   aved By   Date
X _   _ EMERGEN                                                    SPCNTRL 20220919
_ _   _ EMERGEN  Enter Scan Library, Object to be Scanned                   20190108
_ _   _ EMERGEN              or PF3 to Exit                        M712    20220912
_ _   _ EMERGEN          Library : ESPDEMO_                                20220322
_ _   _ EMERGEN     Range : DEMO05__ ________ (*) for ALL                  20220526
_ _   _ EMERGEN          Env No : 01                                       20220526
_ _   _ EMERGEN                                                            20220526
        ESPDEMO          Store Lib : ESPDEMO1                      SPCNTRL 20220914
        ESPDEMO                                                    M712    20220914
_ _   _ ESPDEMO  DEMO03   R  Program                          Y    GM712   20220914
X _   _ ESPDEMO  DEMO04   R  Program                 17   2    Y    ESPCNTRL 20220919
_ _   _ ESPDEMO  DEMO05   R  Program                          Y    GM712   20220914
  L=Lib                      ** Start of Data **
   Restart at Library: ________ Object: ________        No of Objects: 2
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Dupl  AddCC Quit  Error ScanD Scan        PgDn        Curr
```

PF5 – Scanning Modules in Development libraries adding to another library

This function (**PF5 – Scan Devlibs**) is a combination of PF6 and PF1 scanning functions, whereby the module is scanned in the Environment number entered (a Development Environment), all object information is obtained from the scan, but the module is stored in the EspControl Inventory using the Store Library as reflected in the figure above. The difference to PF1 is that it does not make a duplicate copy of the code in the Store Library.

The validation rules for PF5 are as follows:

- The Library entered will be used for the actual scan and must exist in a Development environment.
- The module range will be used for the scan and these modules must exist. No entries will be created if the modules do not exist (if this is what is required – use PF2).
- The Store Lib will be validated against the STORECC code in CC010 and must have a valid entry in STORECC for PF5 to work, else an error will be returned.
- The scanned in objects will be created in the Inventory List, but under the Store Library and NOT the scanned in Library.

3.2.5 CC054 – Copy an ISR

This function is used to copy the links of a completed ISR to a new object, instead of manually linking the objects via CC050 one by one.

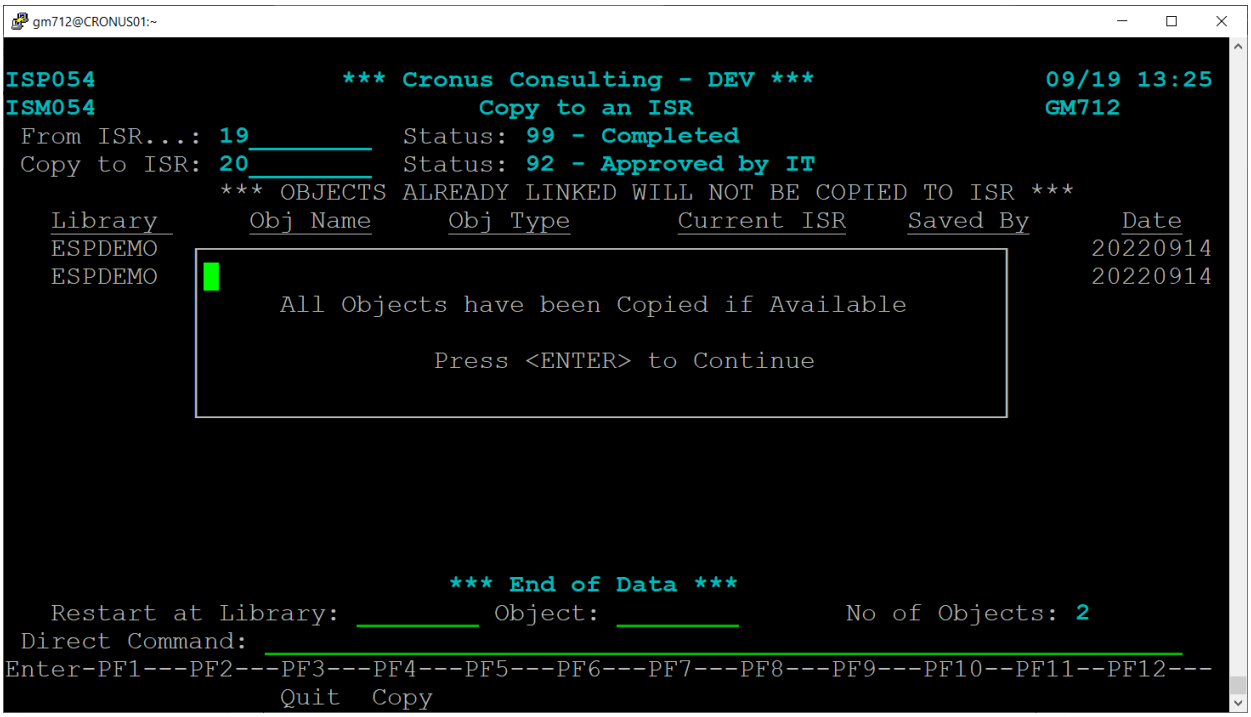
```
gm712@CRONUS01:~

ISP054                 *** Cronus Consulting - DEV ***                09/19 13:25
ISM054                          Copy to an ISR                        GM712
 From ISR...: 19________   Status: 99 - Completed
 Copy to ISR: 20________   Status: 92 - Approved by IT
             *** OBJECTS ALREADY LINKED WILL NOT BE COPIED TO ISR ***
   Library      Obj Name      Obj Type       Current ISR    Saved By       Date
   ESPDEMO      DEMO02        Program                       GM712       20220914
   ESPDEMO      DEMO03        Program                       GM712       20220914

                              *** End of Data ***
   Restart at Library: _______ Object: ________        No of Objects: 2
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit  Copy
```

Copy an ISR and link objects to a New ISR

The ‘COPY FROM’ ISR must be a completed ISR. The ‘COPY TO’ ISR must be a new ISR that has been created via CC030/CC035 with a linked profile and then, if required, approved via CC040. The new ‘COPY TO’ ISR must not have had any migrations or objects linked, else the copy will not be allowed.

The objects that were linked to the Copy From ISR will be displayed on the screen, including the Current ISR No, if any. If any objects are already linked to another open ISR, they will be displayed for the user to see, but will NOT be copied to the new ISR.

Press PF4 for the linking of the available objects to the new ISR.



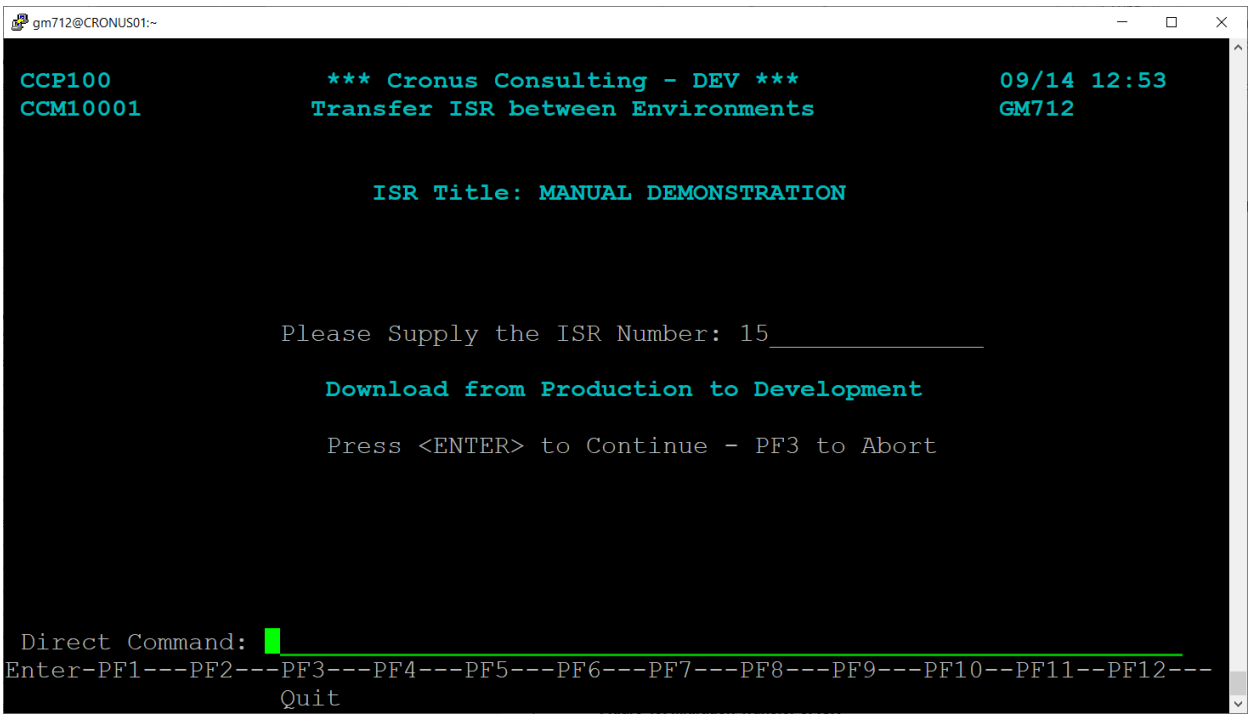
PF4 Confirmation of Copied Objects to new ISR

Once PF4 has been pressed and the objects have been copied to the new ISR, the transaction is complete. To view or de-select any of these objects, the user must once again go and use function CC050. Once in CC050 objects may be removed and others may be added. CC054 is a linking tool only and has no other function.

3.2.6 CC100 – Transfer an ISR between Environments

The function is used to **transfer** all **objects** linked to an ISR between the defined environments. Any user that has access to this function can transfer an ISR, Additional security is controlled with **CC060** – **Upload Approval** per environment for the ISR and **USERID** or **GROUP** linked to the profile in **CC002**. If entered, only that user will be allowed to migrate to the specified environment. If ISR SECURITY is set on in CC030, only the originator may transfer objects.

According to the current status of an ISR and the profile linked to it, the system will determine to which environment objects must be transferred. Those already in Development (New Objects) will not be transferred until all other objects have been transferred to Development from the Master Environment of that profile if the profile specifies a download move.



Migration Transfer of ISR

If Upload Approval is required by the specified profile, it must first be APPROVED in CC060 before the migration will be allowed. However, the migration WILL always be DISALLOWED if the Return Approval indicator has been set on. This will ensure that an ISR cannot be both Transferred and Returned at the same time. If Return is no longer required and objects have been marked for return, use PF5 to reset objects in function CC053.

Once the user has pressed enter to continue, the ISR will be put through the PATH VALIDATION routine, if the VALIDPATH code is set to Y (Yes) in CC010. This routine will validate all LUW directories/paths required (fuser, source staging, backup paths set up in the Control Variable section CC001), as well as

testing the existence of objects in the FROM environment to ensure unnecessary ISR failures and consequent restores. If the Path Validation fails, the ISR will be put on HOLD for investigation and once the paths or objects are in the correct state, the user may reset the ISR via CC075 and redo function CC100. The Path Validation for the existence of Objects in the From environment is completed in full, and then if any error objects exist, the ISR will fail. The user will be able to print an error report if required or the user may enquire on all error objects in the ISR History transaction (CC095). The TO environment LUW paths (as specified in CC001) are also checked and if they do not exist, they will be created in LUW via the path validation routine. If this is the case, the path validation routine will run longer than usual. ESPERRTA is also checked for existence in library SYSTEM in both the From and To environments and will return an error and the ISR will fail if ESPERRTA does not exist in the required environments.

If “SOURCE MOVE” as part of the linked profile, and the master control profile in CC001 has been marked with N or M, then the existence of objects will be checked. Since the environment is one where there should be no source, if source does exist it will create a problem and will require user intervention. See below in manual for further re-explanation.

NON-COMPILE in Initial Environment during DOWNLOAD from Master

When transferring objects from the Master Environment to the Initial Environment e.g. Prod to Dev and the object does not compile, the ISR is not stopped with an automatic restore. The ISR will continue and store all Error Compile Objects in the ISR History. This ISR History transaction (CC095) will allow the user the opportunity of checking out the details of the non-compiles. The Object that does not compile in the Initial Environment is transferred as source only and the user will have to manually compile these objects in this environment before continuing with the ISR. The user will also be able to print a non-compile error report, if required.

Please note, this non-compile ONLY exists in the Initial Environment specified in the linked profile, when DOWNLOADED and when the object is again migrated in a UPLOAD transfer, a COMPILE by the user will be necessary for the successful migration to the next environment. If the RESTORE-IND indicator is set to ‘N’, which implies no restores if errors occur, this option will override the above changes as the ISR is not restored in any event due to compile errors when downloading.

NON-COMPILE Error Report

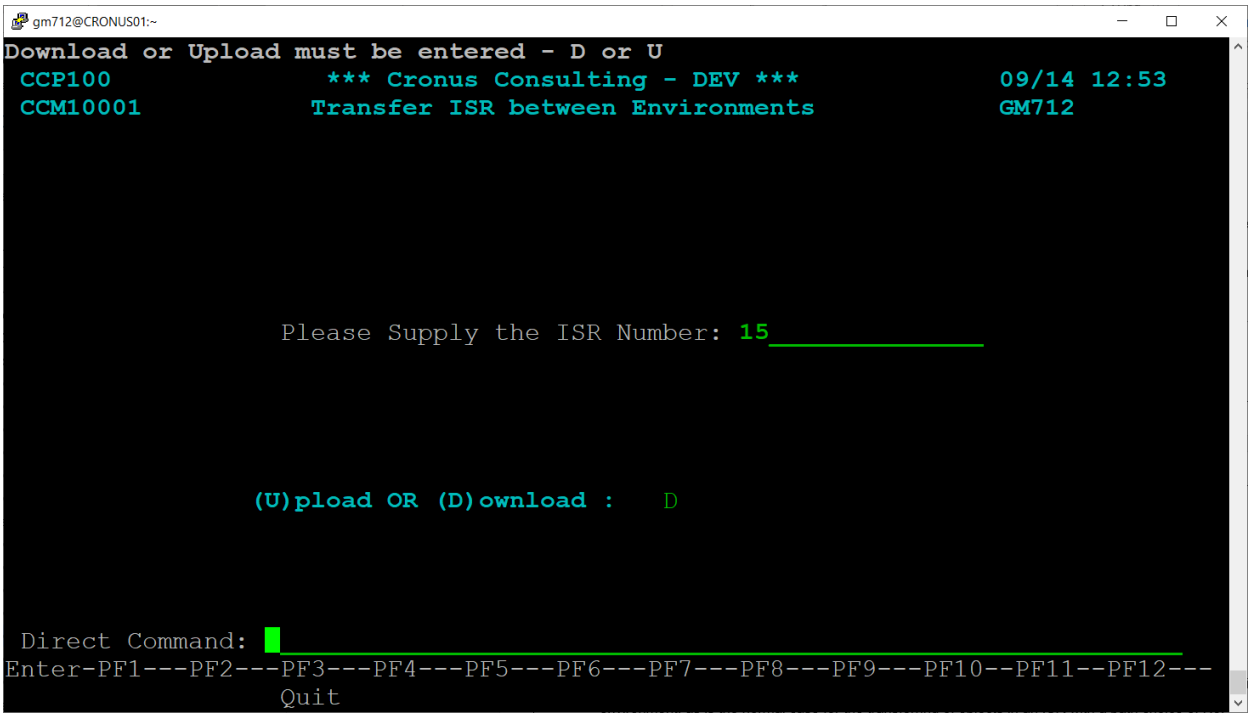
If the RESTORE-IND = ‘Y’ and the above condition occurs, then the user has a choice to print out an Error Report reflecting all objects in Compile Error. The user will be requested to input a “Y” if a report is required and “N” if it is not. If YES, the report will be sent to the relevant print queue.

```
gm712@CRONUS01:~
Job number 1239177: N.20220914.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/14                    Transfer ISR Compile Error Report - ISR No -      >
14:02:12.6                     http://www.cronus.co.za

Environment From: Production
Environment To:   Development


Program    User    Date        Time                                Error Descri>
   Id       Id
-------- -------- ----------------- ------------------------------------------>

CCP100   GM712    20220914 14:01:43 From Environment Production to Development >
         GM712    20220914 14:01:43 ESPDEMO DEMO01 - Compile Error - Error No 0>



   Before continuing, Compile Above Programs in Error in Development
   Only SOURCE has been transferred, no object code exists

   If the above is not done, the next UPLOAD for this ISR will Abend
```

ISR Compile Error Report

NON-COMPILE during UPLOAD from Initial Environment

All objects are automatically re-compiled in the Initial Environment, except if the STOW/CAT indicator is marked a N (none) in the linked profile, before being transferred to the next environment during an UPLOAD phase. The marking of NONE in the initial environment will also only move the code in a DOWNLOAD movement from the Master to the Initial environment with a SAVE function only. This means that the NON-COMPILE Error report and routine will not be part of a transfer, as the compile step will be by-passed. If NONE is selected, then it is up to the user to ensure that the source is stowed in the Initial Environment, as if it is not and an error does exist, the source will not stow in the next environment causing the ISR to be restored if the restore indicator is set to Yes.



Download or Upload Window

If the PATH CHOICE of the linked profile is set to YES, then the user will be given the choice of whether the initial movement of the ISR should be a DOWNLOAD or an UPLOAD, as shown above. This choice will only be given at the start of a transfer i.e. the very first time a specific ISR is being transferred

If D for DOWNLOAD is taken, the object will be downloaded from the MASTER environment to the Initial environment, as is the normal case for the transferring of objects in an ISR with a path choice of No.
If U for UPLOAD is taken, the movement down from the MASTER environment is skipped and the object will be moved directly up from the INITIAL environment. The objects MUST therefore already exist in this environment, as if they do not the PATH ALIDATION routine will abort and the ISR will be stopped.

If the PATH CHOICE of the linked profile is set to U (Upload), then the user will NOT be given the choice of whether the initial movement of the ISR should be a DOWNLOAD or an UPLOAD, but the ISR will be forced in an UPLOAD movement. With this profile, the object will NEVER be downloaded from the MASTER.
If the objects linked in the ISR are NEW OBJECTS, there will be no window displayed, even if the PATH CHOICE is set to Y/U, as the objects MUST be moved up the line.

Path Validation

```
gm712@CRONUS01:~
Executing ...
 CCP100              Transfer ISR between Environments            09/14 13:30
 CCM10002                                                         GM712
   To        ISR NO: 15 - MANUAL DEMONSTRATION                     NAT
 Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp

            Path Validation for Objects in ISR 15
        ISR will be Rejected in HOLD if any errors occur
            Please wait while VALIDATION takes place

 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Path Validation

If the code VALIDPATH in CC010 is set to N (No), then the path validation is skipped.

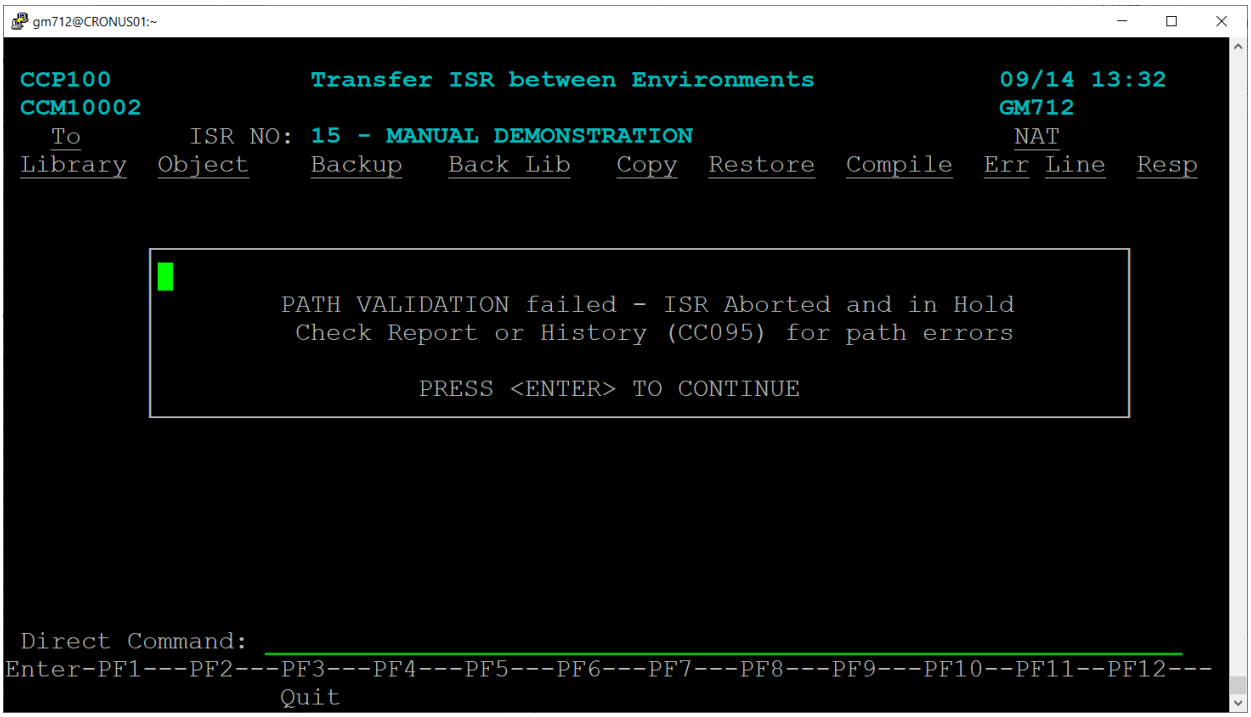
The PATH VALIDATION as seen above will be executed if set to Yes, to check that all objects being migrated exist and that all backup, staging areas, libraries and fuser paths are also valid. In this manner, most of the errors of an ISR will be handled and the ISR will not fail in the middle of the migration causing the entire ISR to be backed out. A permissions error will not be determined in this validation as this will only be found out, when the migration function tries to update a path. If the path validation fails, a window will be displayed as seen below, and the user will be able to go and correct all the errors. The history function (CC095) may be viewed to get the path error and also error objects that do not exist. A report of all error objects, if Y is entered by the user, will be sent to the print queue. The ISR is also put on hold, so no further migration will take place until the problem has been solved. The ISR must be removed from hold (CC075) before being migrated again. Remember, this path validation will also automatically check for program ESPERRTA in library SYSTEM.

```
gm712@CRONUS01:~
CCP100              Transfer ISR between Environments            09/14 13:32
CCM10002                                                         GM712
  To        ISR NO: 15 - MANUAL DEMONSTRATION                     NAT
Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp

        Error Objects Exist - See REPORT for list or CC095

                Enter 'Y' to Print            Y
                     Press <Enter> to Continue

 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Path Validation Error Window

```
gm712@CRONUS01:~
Job number 1186529: N.20220914.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/14         List of Objects with INCORRECT PATH - ISR No -              >
13:33:34.9                    http://www.cronus.co.za

Path Validation Environment : Production          On Server : CRONUS01
     Library   Object   Env     Library   Object   Env
                        No                         No
     -------- --------  ---     -------- --------  ---

     ESPDEMO  DEMO01      3



   These Objects were not found in the FUSER for the Environment
   FUSER PATH /opt/softwareag/Natural/fuser_natp
   Directory for SRC

   Determine why, reset ISR from Hold (CC075) and rerun
   Either ensure correct path or unlink Object from CC050



                          ----- END OF REPORT -----
```

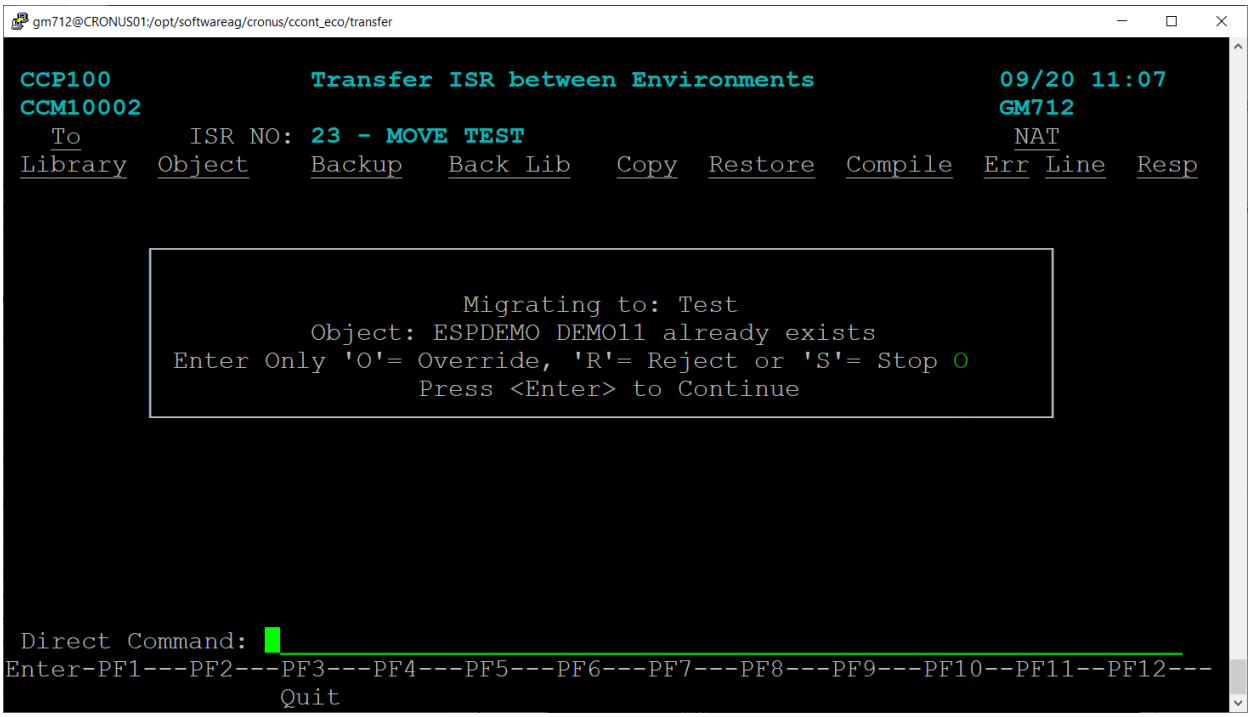
List of Objects with Incorrect Path



Final Confirmation window for Errors in Path Validation

Testing the Existence of Objects in a MOVE SOURCE environment

If the Allow-Source option of the “from” or the “to” environment = M (including on CC001 as this is the controlling indicator for this check), which is REMOVE SOURCE once the migration to the next environment is successful, the existence of these objects is tested. If they exist in the “to” environment, then a window as reflected below will be displayed and the user must decide on one of the following options reflected below the example screen. However, this test will be omitted, even if allow-source is marked as ‘M’ if the Allow-Source indicator in CC001 (master control variables) is marked as a ‘Y’ or ‘N’. In this manner, a site may bypass this check if necessary.



Test Existence of OBJECTS window

The user has 3 choices:

1. **Overwrite the Object** – Copy the linked object to the next environment and overwrite the code that currently resides in the next environment.
2. **Reject the Object** – Ignore the object in this section of the migration transfer and write a record to the History function (CC095) stating that the object has been “Linked but not Copied”
3. **STOP the entire ISR** – This will return the user to the menu with the ISR being placed in HOLD. Once the user has decided on what to do, the ISR must be removed from hold via CC075 and then it can again be transferred via function CC100.

The above options are for one movement of an ISR and on the next movement, the objects will be re-tested, including the rejected ones. If these rejected objects are no longer required to be part of the ISR, they must be unlinked via function CC050, before CC100 can be rerun for the specified ISR.

When compiling objects in the initial environment (if the STOW/CAT indicator ne N (none), the objects will be compiled in the necessary correct order of Natural object types.

```
gm712@CRONUS01:~
Executing ...
 CCP100              Transfer ISR between Environments            09/14 13:36
 CCM10002                 From Development To Test                 GM712
   To        ISR NO: 15 - MANUAL DEMONSTRATION                      NAT
 Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp

 ESPDEMO  DEMO01          F  ESPDEMO    Y  3    N       S  3
 ESPDEMO  DEMO08          F  ESPDEMO    Y  3    N       S  3
 ESPDEMO  DEMO09
 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Current Migration Details of ISR

The following attributes are displayed for each object transferred:

- To Library - Library that Object is being migrated TO. This will be either the current library that the object is in, or the To-Library selected in CC050 library option where the object is being migrated to or the ISR Library updated in the Profile record or the GENERIC library rule from ISNUX100.
- Object - Name of Object
- Backup - “T” indicates that a backup copy was taken in the TO environment
  “F” indicates that a backup copy was taken in the FROM environment
  “N” indicates that NO backup was taken,
  “E” indicates there is an error with the Backup and a report will be sent to Printer Queue at end of ISR of all objects not backed up if No restore option taken or view CC095
  See function CC002 for the backup cycle of the migration.
  The duration will be reflected in between the two backup indicators.
- Back Lib - Backup Lib is the Lib of the Object being backed up in the FROM env.
- Copy - “Y” indicates that the object was copied and the duration.
  “E” indicates there is an error with the Copy (and report will be printed on if restore indicator is No)
- Restore - “N” indicates that the download operation was completed successfully.
  “Y” indicates that the object was restored to the backup version, depending on if one exists for the current ISR, or the previous backed up

version. If the restore indicator is NO, then N will mean that no restore will be done if any object is in error

- Compile - "S" indicates that the object was compiled with a STOW, "C" indicates that the object was CATalogued, and "N" indicates that the object was moved and SAVED only and the duration of the compile time depending on STOW-CAT indicator set on profile.
  "E" indicates there is an error with the Compile (and report will be printed only if restore indicator is No or the move is to the Initial Environment).
- Nat Error - Contains the Natural error and line number if a compile error occurred
- Response - Response Codes 11 and 20 and Blank indicates successful completion, else all other response codes indicate a problem.

Once an ISR is complete, without any errors, a transfer confirmation window will be displayed.

If any error should occur during the transfer process, and the backup and restore indicators are set to Yes, all objects that form part of the ISR will be restored to either the prior backup version or the current version if the backup has already been taken in this ISR. If the restore indicators are set to No, the ISR will continue with an error report printed at the end of the transfer. **If the user requires a restore of the previous ISR and not of the current version, this may be done during a migration process via function CC200 to any environment and the library of choice.**

If RESTORE INDICATOR = Yes, then if any error should occur during the transfer process, all objects that form part of the ISR will be restored as specified above to the environment it is being moved from.
If RESTORE INDICATOR = No, then an object in error will reflect "E" in the correct column and the ISR migration will be continued with the other objects. NO restore of the error objects will be done, but a RESTORE ERROR report will be sent to the print queue if required and requested by the user at the end of the current migration function. If not printed, these errors may also be viewed in the History enquiry (CC095).

```
gm712@CRONUS01:~

CCP100              Transfer ISR between Environments            09/14 13:08
CCM10002              From Production To Development             GM712
  To        ISR NO: 15 - MANUAL DEMONSTRATION                     NAT
Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp


            Downloaded from Production to Development

                 Press <Enter> to Continue




ESPDEMO  DEMO01          N             Y  3    N       S  1
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

Transfer Confirmation without any errors

See example windows below for transfer confirmation with no restore, error with an ISR, or error with NO restore, but restore printout. The Restore window is the same for all restore functions and a sample of this window may be viewed in the explanation of CC110 in this document. If there is no backup of the object at all, whether via the current ISR or a previous version, no restore of this object will be done and a sample of this window may also be viewed in CC110.

```
gm712@CRONUS01:~
CCP100              Transfer ISR between Environments            09/14 13:47
CCM10002                 From Development To Test                GM712
  To        ISR NO: 15 - MANUAL DEMONSTRATION                     NAT
Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp

        Compile Error: Objects Restored if Backup= Complete
         All Objects Restored if Backup Indicators correct
                  Press <Enter> to Continue

ESPDEMO  DEMO01          F  ESPDEMO    Y  3    Y       S  3
ESPDEMO  DEMO08          F  ESPDEMO    Y  3    Y       S  3
ESPDEMO  DEMO09          F  ESPDEMO    Y  3    Y 14    S  3   0082 0020
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

Transfer ISR with ERROR

```
gm712@CRONUS01:~
CCP100              Transfer ISR between Environments            09/29 07:45
CCM10002                 From Development To Test                GM712
  To        ISR NO: 33 - NEW ISR COMBINE                          NAT
Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp

                Uploaded from Development to Test

            Print Restore Error Report            Y
                Press <Enter> to Continue

EMERGENC ZGL13           F  EMERGENC   Y  4    N       E  4   0623 0010
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

ISR in Error with No Restore

These objects are given the same ISR STATUS as successful objects and therefore these errors must be fixed manually, as the next time that CC100 is run, it will assume that these objects are correct in the same environment as all other successful objects in the migration. If not needed, unlink via CC050.

```
gm712@CRONUS01:~
Job number 1278070: N.20Undefined key sequence - ignoredRINT                START
.....30...35...40...45...50...55...60...65...70...75...80...85...90...95...100..
     Transfer ISR Error Report - ISR No -                        33   Page:    >
      http://www.cronus.co.za


tion


      Time                                Error Description                  >

----------- ------------------------------------------------------------------>

29 07:46:06 From Environment Development to Test                             >
29 07:46:06 EMERGENCZGL13 - Compile Error - Error No 0623 Line 0010 - No Restor>
29 07:48:46 From Environment Development to Test                             >
29 07:48:46 EMERGENCZGL13 - Compile Error - Error No 0623 Line 0010 - No Restor>
29 08:00:41 From Environment Test to Production                              >
29 08:00:41 EMERGENCZGL13 - - No Restore Done                                >
29 08:00:41     Error Message - 05 - FUSER - Source Code does not exist      >



     ----- END OF REPORT -----
```

Sample ISR Error Report with No restore

If BACKUP of any particular object does not complete, and the restore indicators are set to No, the migration will not abort, but will reflect a window stating that a non-backup report has been sent to the print queue. This report contains all objects that were not backed up. Users may determine the reason, and fix if necessary. Return non-backed up objects via CC102 and then re-migrate via CC100 to get a correct backup. If a new object is transferred, then a backup will not be taken in the TO environment as the object never existed in the Master environment. The backups get taken AFTER amendment and at the start of a move to the new environment, so new objects and existing objects will be backed up in the FROM environment, except when moving to Production.

If the BACKUP fails and the From or To Environment is the Master Index and the restore indicators are set to Yes (with the exception of new objects), then the ISR will abort and be restored as explained in a transfer error above.

```
gm712@CRONUS01:~
Job number 1287776: N.20220929.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/29                    Audit Report for transfer of ISR No - 34          >
08:09:50.7

ISR Title....: TEST BACKUP              Xref Number....:                        >
               TEST BACKUP                                                      >

Environment From: Test
Environment To:   Production

LINE NO    TO      OBJECT  BACK TO BACK FROM  BACKUP  COPY RESTORE COMPILE  NAT >
        LIBRARY     NAME     IND      IND      LIB    IND    IND     IND   ERROR>
------- -------- -------- ------- --------- -------- ---- ------- ------- ----->

0001               Upload from Test to Production

0002    EMERGENC ZGL13    E       F         EMERGENC                          >

0003    All Objects Restored if Backup Indicators correct                     >


                              ----- END OF REPORT -----
```

ISR Backup Error Report

If the Backup is being done during the TO Migration and the To environment is the MASTER environment, then the ISR will abort if the backup was not successful, and the ISR will be restored according to the profile rules. See example window below.

```
gm712@CRONUS01:~

CCP100              Transfer ISR between Environments            09/29 08:09
CCM10002                 From Test To Production                 GM712
  To        ISR NO: 34 - TEST BACKUP                              NAT
Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp


        Backup To True Master Env Error: Restored if Applicable
           All Objects Restored if Backup Indicators correct
                    Press <Enter> to Continue




EMERGENC ZGL13     E     F  EMERGENC          Y  2
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                Quit
```

ISR Backup Error to Master Environment

AUDIT REPORT – if AUDITRPT is set to Y in CC010 then see examples below of the printed report

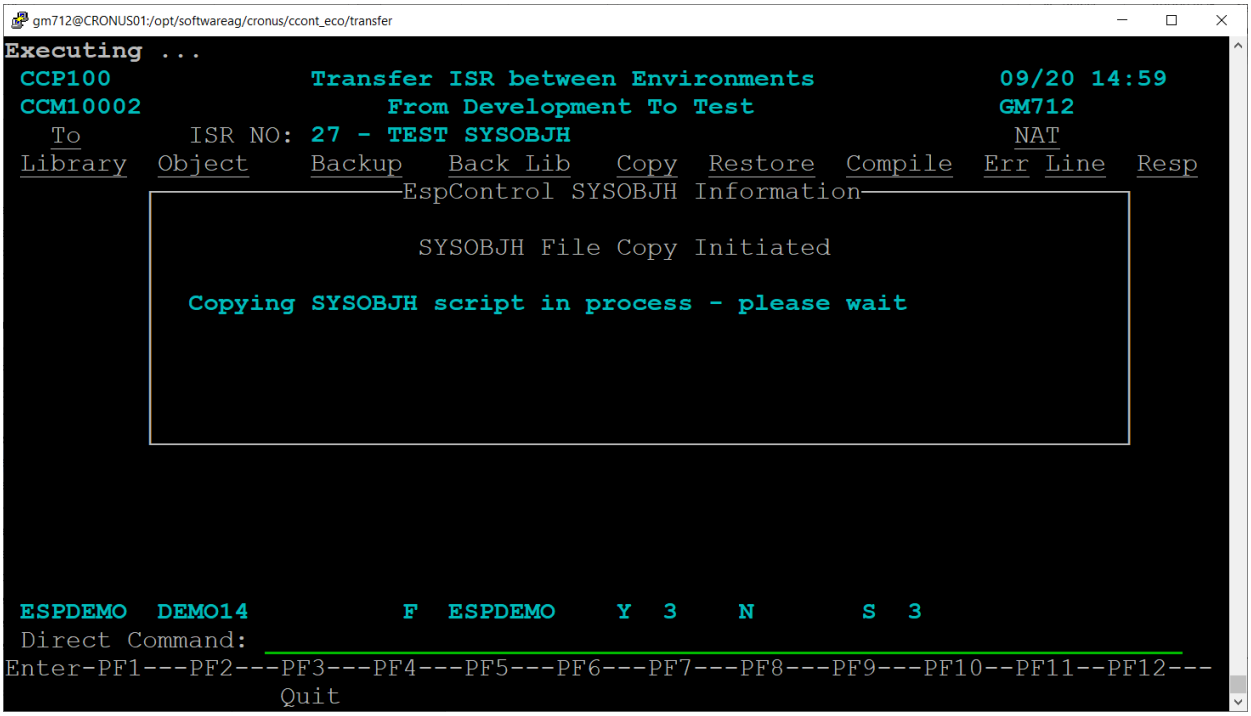
```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer
Job number 1133181: N.20220920.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/20          List of Objects with INCORRECT PATH - ISR No -             >
08:39:37.8                     http://www.cronus.co.za

Path Validation Environment : Development          On Server : CRONUS01
     Library    Object    Env      Library    Object    Env
                          No                           No
     -------- --------    ---      -------- --------    ---

     ESPDEMO   DEMO11       1



   These Objects were not found in the FUSER for the Environment
   FUSER PATH /opt/softwareag/Natural/fuser_natd
   Directory for SRC

   Determine why, reset ISR from Hold (CC075) and rerun
   Either ensure correct path or unlink Object from CC050



                              ----- END OF REPORT -----
```

Audit Report with Errors

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer
Job number 1484999: N.20220920.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/20                   Audit Report for transfer of ISR No - 27          >
15:01:18.8

ISR Title....: TEST SYSOBJH            Xref Number....:                        >
               test sysobjh                                                    >

Environment From: Development
Environment To:   Test

LINE NO    TO      OBJECT  BACK TO BACK FROM  BACKUP  COPY RESTORE COMPILE  NAT >
        LIBRARY     NAME     IND      IND      LIB    IND    IND     IND   ERROR>
------- -------- -------- ------- --------- -------- ---- ------- ------- ----->

0001             Upload from Development to Test

0002    ESPDEMO  DEMO14           F         ESPDEMO  Y    N       S            >

0003    SYSOBJH Download completed successfully in Env 1                       >

0004    Uploaded from Development to Test - SOURCE - STOW - BACKUP             >
```

Clean Audit Report

SOURCE UNLOAD function

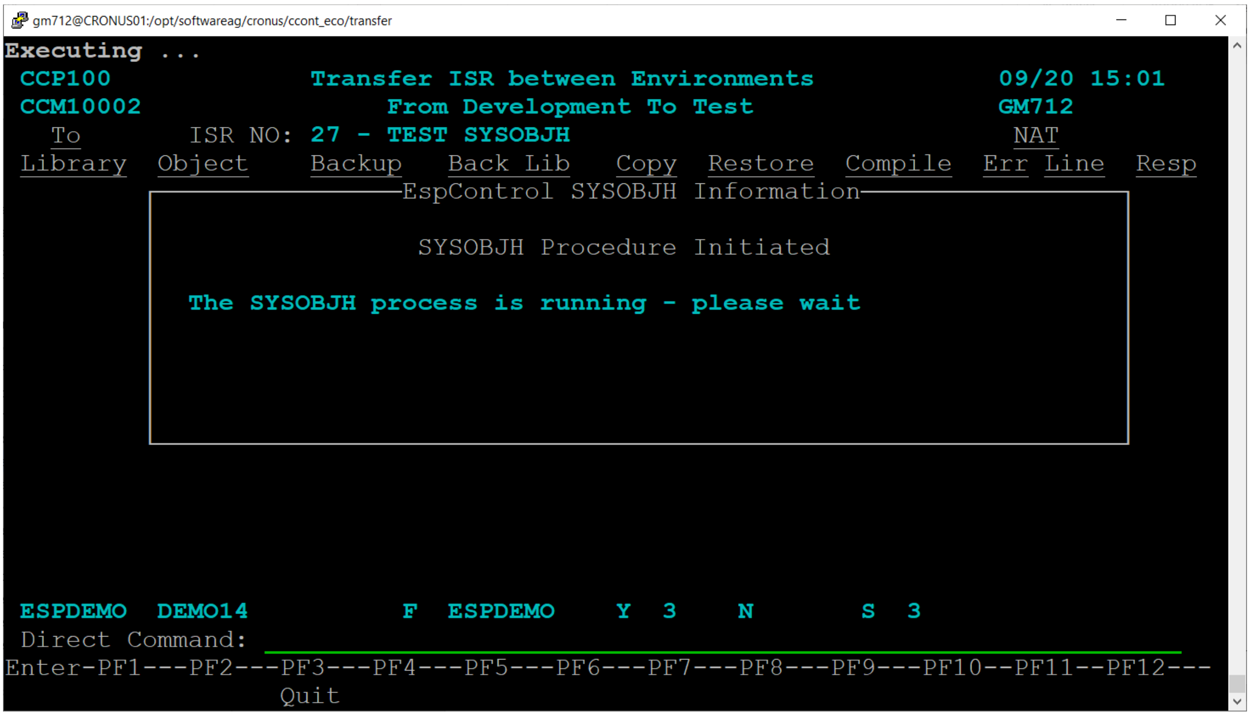


Start of SYSOBJH Function

If the SOURCE UNLOAD indicator has been set to ‘Y’ in the linked Profile, the SYSOBJH function to unload the source during the migration of the FROM environment, as a text file on a LUW platform, will be called at the end of the successful migration of the ISR.

If the linked profile has a RESTORE INDICATOR of NO, then the SYSOBJH function will be called even if some objects were not successfully migrated.

This function will unload all the source of all objects that have been migrated to a specified /transfer directory on a LUW platform. This directory has been built up from the shell script path in CC001 with /transfer appended. This function will create a report saved on the LUW server, as well as a text file with the ISR number as the start of the file name. The time will be appended to the files, in the case of a migration being returned to the previous environment or new objects being linked. This new file will only contain the objects being moved once again, and not the objects that have already been successfully migrated to the next environment in the specified ISR. Therefore, more than one file for a migrated ISR could exist and all these files will have to be used in the Upload of the source via SYSOBJH if required, as they will reflect different objects at different migration times.



```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer
Executing ...
 CCP100              Transfer ISR between Environments          09/20 15:01
 CCM10002                 From Development To Test              GM712
   To          ISR NO: 27 - TEST SYSOBJH                         NAT
 Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp
                  ----------EspControl SYSOBJH Information----------

                         SYSOBJH Procedure Initiated

            The SYSOBJH process is running - please wait


 ESPDEMO  DEMO14          F  ESPDEMO    Y  3    N       S  3
 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Current SYSOBJH Function

See screen below for examples of text files and the subsequent file names that would have been generated are as follows:

- **27.sysobjh.report.1452549** where 27 is the ISR number and the timestamp is at the end of the name
- **27.sysobjh.src.txt.1452549**

The above files are created in the /transfer directory using the SHELL SCRIPT PATH set up in function CC001 depending on the environment move.

The .txt file should be used in the SYSOBJH Upload function. (The export file is created in “transfer” format).

The below screen will be reflected after the successful unload showing the directory where to find the REPORT and TEXT file. The user must now ENTER to finalize the Update of the current movement of the ISR. At this stage the ISR status and the Object Status of each linked object will be updated to ensure that the object has completed the migration successfully.

Copyright © 2026 Cronus Consulting

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_tst
CCP100              Transfer ISR between Environments            09/20 14:52
CCM10002                 From Development To Test                GM712
  To        ISR NO: 27 - TEST SYSOBJH                              NAT
Library  Object    Backup   Back Lib   Copy  Restore  Compile  Err Line  Resp
                  EspControl SYSOBJH Information
                 SYSOBJH Procedure Executed:
Unload command executed successfully. Report is in:
  /opt/softwareag/cronus/ccont_eco/transfer/27.sysobjh.report.
     Press <Enter> to Continue with Final Update of ISR
ESPDEMO  DEMO14          F  ESPDEMO    Y  3    N       S  3
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

SYSOBJH Confirmation window

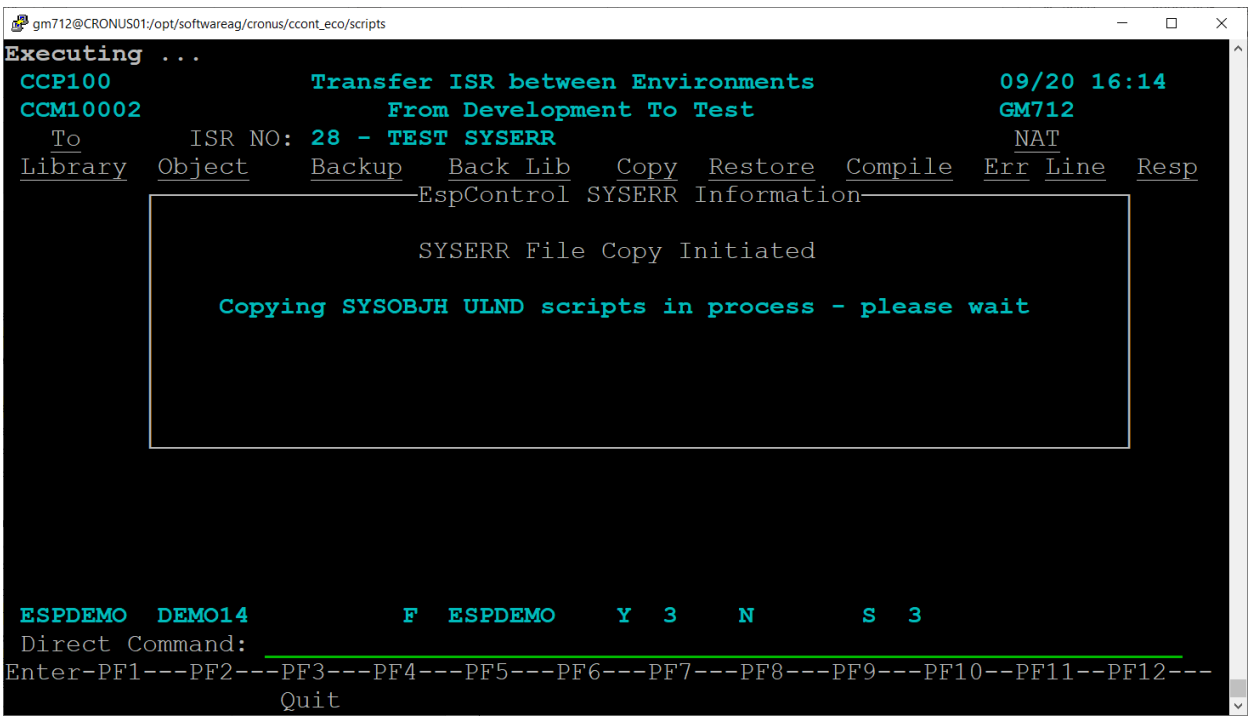
```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer
Job number 1472280: 27.sysobjh.report.1452549 on form type standard        START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
                 *** Unload Objects ***
             Processing TRANSFER Work File $NATWK01

Library  Object Name                      Type        S/C  DBID/FNR   Date     >
-------- -------------------------------- ----------- --- ----------- --------->
ESPDEMO  DEMO14                           Program     Src    22/31    2022-09-2>

Function completed successfully.

                          --- End of file ---
```

SYSOBJH Unload Report

SYSOBJH and SYSERR errors

The transferring of system error messages, also uses the SYSOBJH unload routine and if this routine aborts for either the Source Unload or System Error Message function, the rules for this error are the same in both SYSOBJH and SYSERR functions. Therefore this error routine, how to fix and continue with the migration, is only explained using the example in the window reflected for the system error messages. If an error does occur in the source unload transfer, the SYSOBJH ERROR indicator is set (see function CC076) and all other ISR functions will be blocked until the source has been unloaded without error.

SYSTEM ERROR MESSAGE transfer

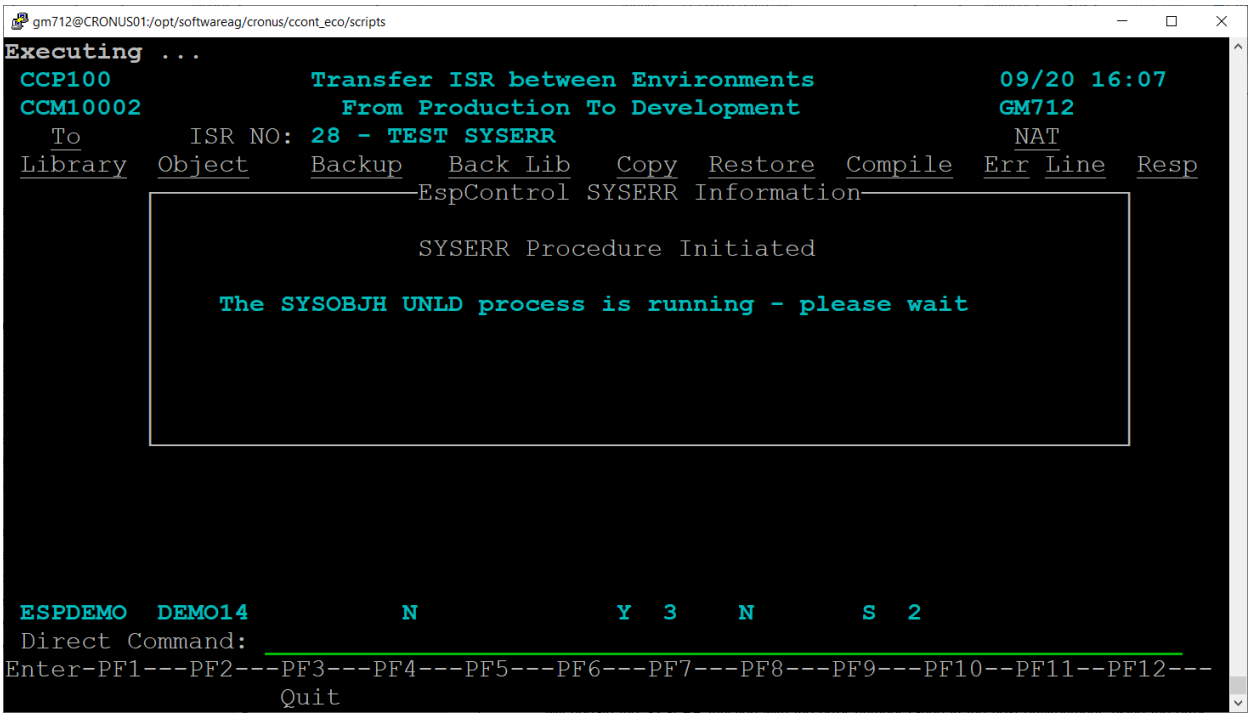


Start of SYSERR transfer

If System Error Messages have been linked to an ISR in CC050, the SYSOBJH function to transfer these error messages during the migration of the FROM environment to the TO environment, will be called at the end of the successful move of all objects linked in the ISR. If the linked profile has a RESTORE INDICATOR of NO, then the System Error transfer function will be called even if some objects were not successfully migrated.

This function will transfer all system error numbers with from and to libraries specified in PF4 in CC050 and will update the SYSERR function with the error number range in the next environment, using the error message data from the previous environment. Both a load and unload report will be created in the /transfer directory. Each time a migration is done the system error messages will also be transferred. So, therefore, if an ISR is returned and then re-migrated the error messages will again be transferred. If an ISR is

downloaded from the Master Environment, then the error messages will be taken from this environment to the Initial Environment including the objects, and then may be amended to move upwards if necessary. **If error messages are only ever migrated in an upload movement i.e. from the Initial Environment upwards, then create an ISR linked to a dummy program only and choose the path of UPLOAD when using CC100, and in this way error messages will always move up the line.**



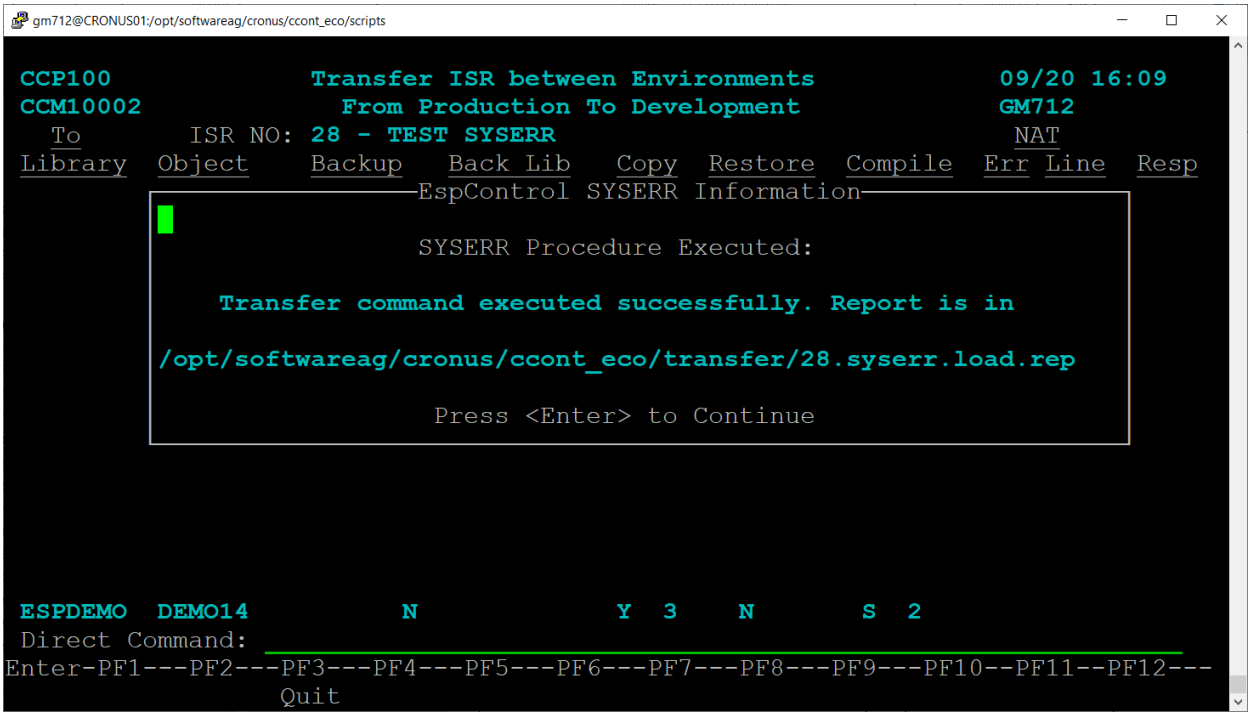
Actual SYSERR transfer

See screen below for examples of text files and the subsequent file names that would have been generated are as follows:

- **28.syserr.unld.report** where 28 is the ISR number
- **28.syserr.load.report**

The above files are created in the /transfer directory using the SHELL SCRIPT PATH set up in function CC001 depending on the environment move.

Once completed successfully, the following screen will be reflected:



SYSERR Confirmation window

The above screen will be reflected after the successful transfer of the error messages to the next environment. The user must now ENTER to either finalize the current movement of the ISR or create the SYSOBJH file if the linked profile has SOURCE UNLOAD = ‘Y’. At this stage the ISR status and the Object Status of each linked object will be updated to ensure that the object has completed the migration successfully, if no SYSOBJH has been called for. If a SYSOBJH unload is still to occur, then this will complete first before updating the object statuses.

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_tst/transfer
Job number 1612901: 28.syserr.load.report on form type standard            START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
                 *** Load Objects ***
             Processing SYSUNLD Work File $NATWK01
             Processing Load File created on 2022-09-20 at 16:35

Library  Error Number  Language Code  S/L   DBID/FNR   Type                   >
-------- ------------  -------------  ---  ----------- ---- ------------------>
ESPDEMO      9000            1        S       22/32     U   User error message >
ESPDEMO      9001            1        S       22/32     U   User error message >

Function completed successfully.

                              --- End of file ---
```

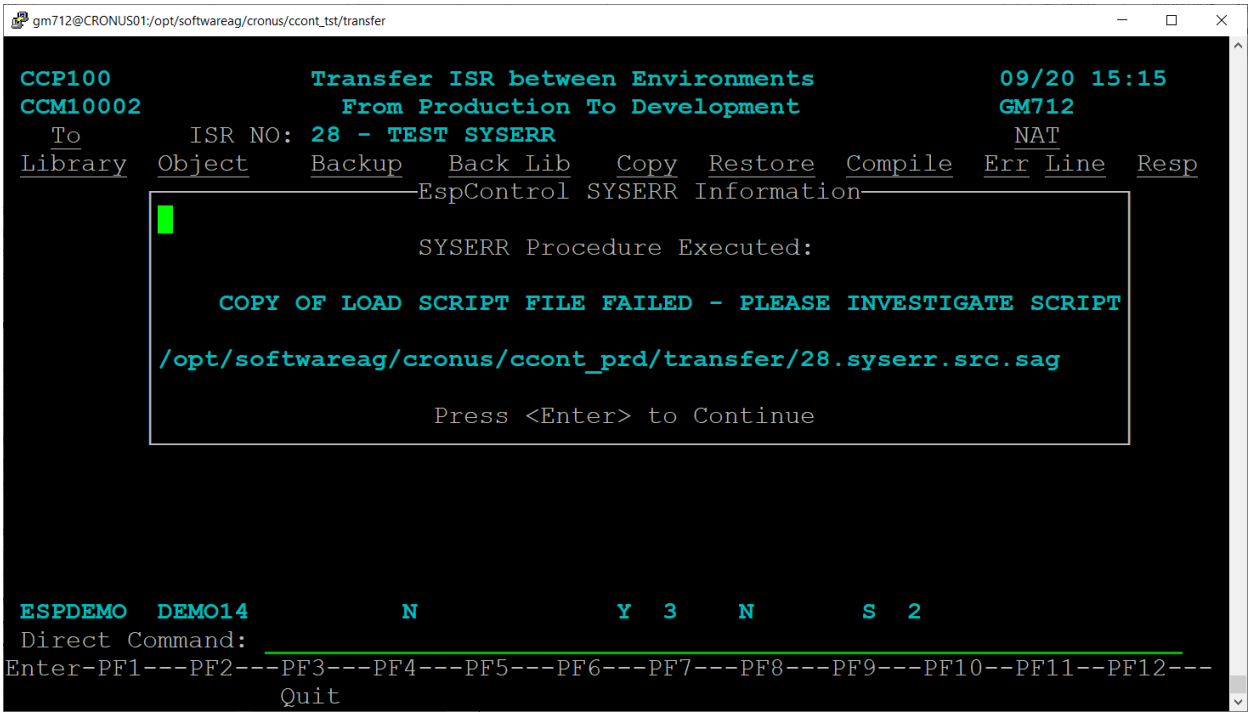
Example of SYSERR report

If the TRANSFER of System Error Messages or Source Unload aborts, the following screen will be reflected with the appropriate error message. In the Source Unload option, the error window and instructions will reflect SYSOBJH.

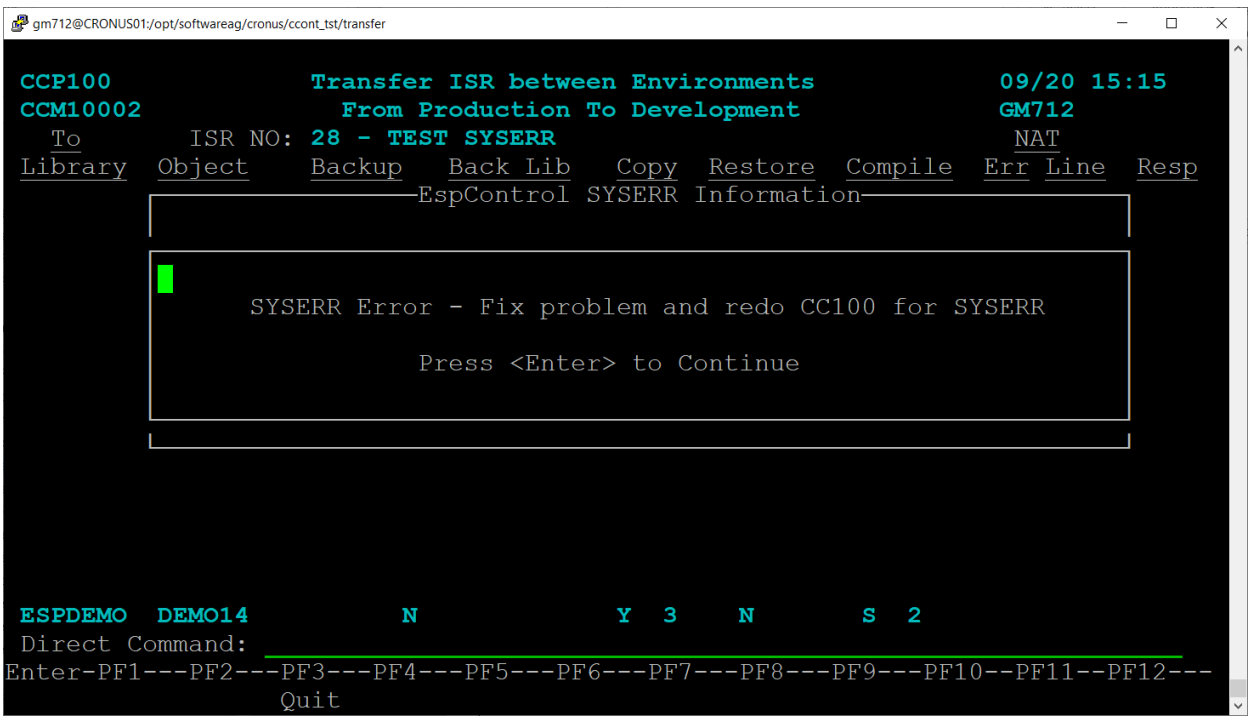
This error message may vary according to the actual error. The logs or the ISR History function CC095 may be viewed to check the error status of the unload.

See next page for example:

SYSERR Error Screen and the following screen - SYSERR and SYSOBJH Error Instructions



SYSERR Error Screen



SYSERR Error Instructions

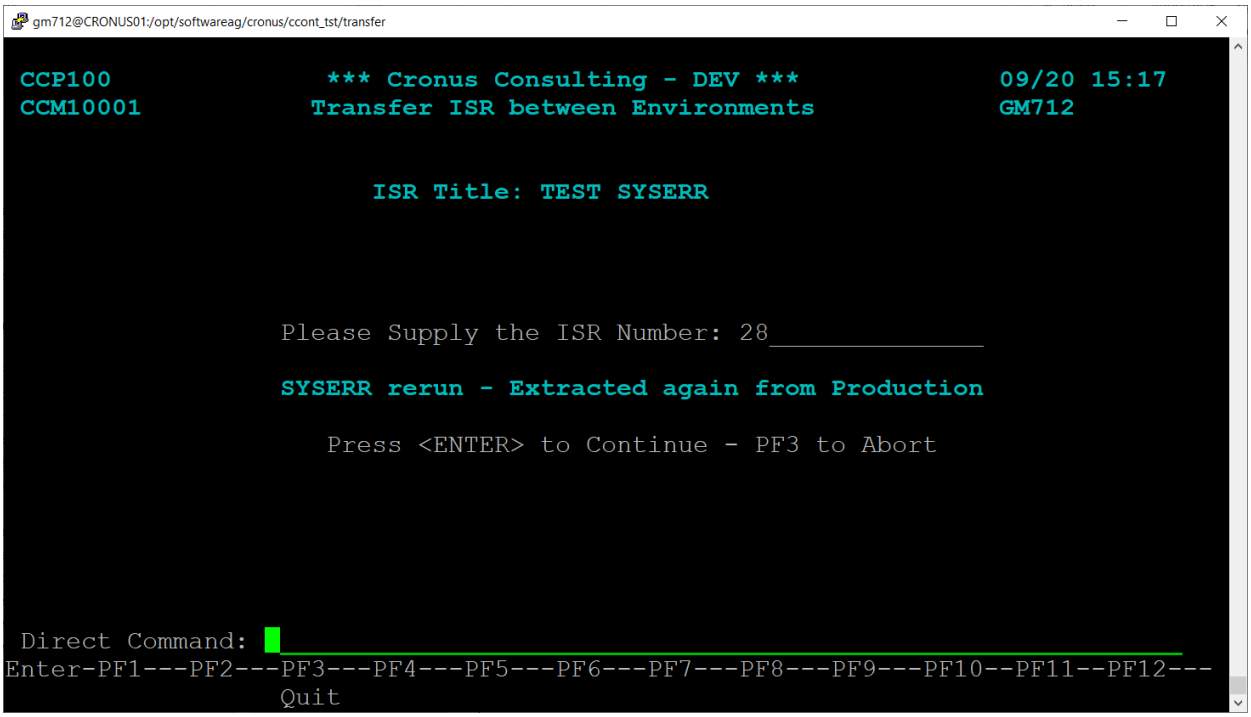
The SYSERR and SYSOBJH source unload transfer function have been set-up in such a manner that an error in the TRANSFER does not RESTORE the entire ISR. The SYSOBJH or SYSERR transfer only commences once the objects have been moved. If an error in SYSOBJH or SYSERR unload occurs, the

user should correct the problem and redo CC100. This will only redo the SYSERR or SYSOBJH Function, but not the migration of the objects. Once the SYSERR or SYSOBJH function is successful, the migration may continue as normal to the next environment.

If an error occurs, the SYSERR ERROR indicator is set (see function CC077) and all other ISR functions will be blocked until the error messages have been transferred without error.

For a SYSOBJH ERROR, another indicator is set (see function CC076) and the same rules as specified above will apply.

Rerun of CC100, for SYSERR Error or SYSOBJH source unload will reflect SYSOBJH



SYSERR Error Rerun of CC100

The above is an example of where a SYSERR error occurred and is now re-running CC100. The above message will be displayed as the objects will NOT be re-transferred, but only the source unload or system error messages transfer, depending on what is appropriate for the ISR.

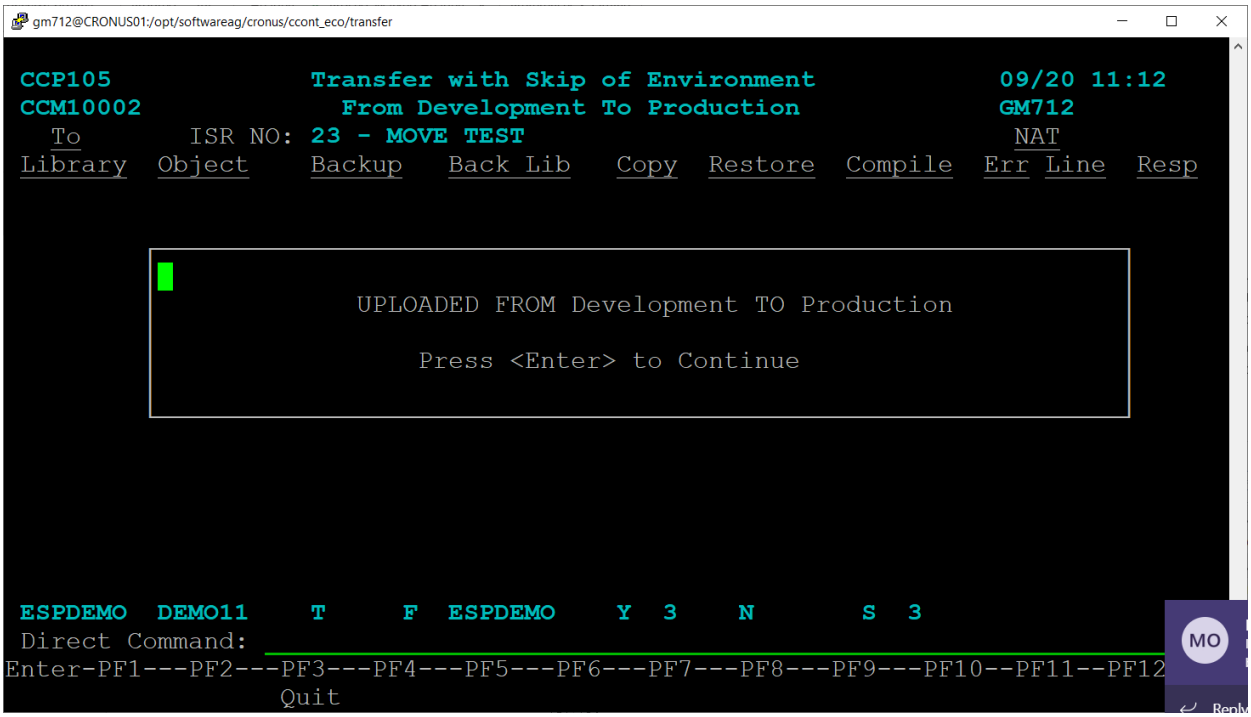
3.2.7 CC105 – Transfer with Skip of Environment

The function CC105, transfer with a skip of an environment, works in the same manner as the CC100 migration function, except that it allows the user to skip an environment. As you can see below the migration to the TEST environment has been skipped and the object will move directly to Production. This means the TEST will not have a backup or the new version of the source, since it was skipped.

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer

CCP105              *** Cronus Consulting - DEV ***            09/20 11:12
CCM10501           Transfer with Skip of Environment           GM712

                       ISR TITLE: MOVE TEST

                       Test will be SKIPPED

                 Please Supply the ISR Number: 23_____________

                    Upload from Development to Production

                   Press <ENTER> to Continue - PF3 to Abort

 Direct Command: _____________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Transfer with Skip of an Environment (direct from Dev to Prod, Test skipped)

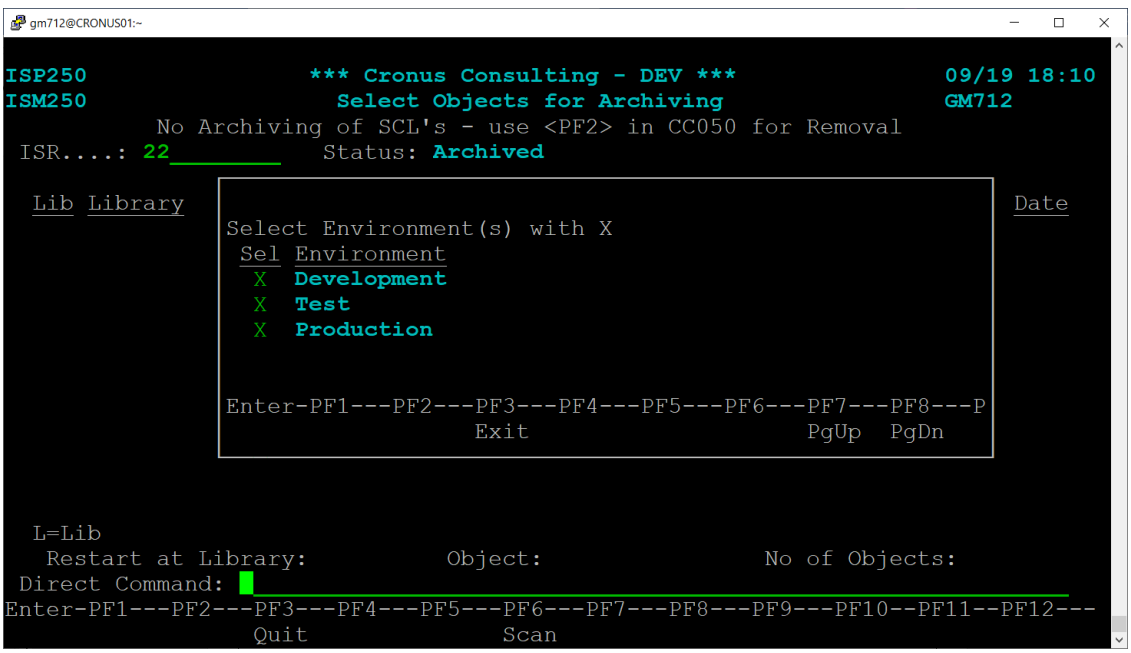
There are specific rules regarding skipping, and there must be at least 3 environments defined. The skip to Development from Production (the initial download move) is NOT allowed. To skip the download move, the profile with a Path Choice of “Y” must be used and the user must request Upload only.



Confirmation window after a SKIP

3.2.8 CC250 – Select Objects for Archiving

This function is used to **remove** objects from the EspControl object **inventory** and the actual module is **scratched** in the Natural **Source** library for all the selected environments. The library that must be used can be selected if required, per object, as in the CC050 selection and will be used for the chosen environment for archiving.



Archive ISR

Enter the ISR number (**ISRTYPE** on CC035/CC040 must be **OLD** to use this function) and select the required environments where the Object must be archived from. All the environments or a subset of these connected to the linked profile will be used. The object will be backed up if available, scratched from the selected environments and removed from the Object Inventory List (CC050). The archive log may be viewed in function CC095. As the program is removed from the Inventory List it will not be viewable from any object enquiries once archived. However, if the object name is known it may be entered in function CC088 using PF6 and then all available history will be displayed, even if this object no longer exists. This could always be manually restored from the backup path specified in the history, if necessary. If the object must again be added to the Inventory List, it may be scanned in CC050 using PF6 and a new object entry will be created.

Select the objects for archiving. Confirm the start the archive process with function key **PF4.**

```
gm712@CRONUS01:~

ISP250                 *** Cronus Consulting - DEV ***                09/19 18:10
ISM250                   Select Objects for Archiving                 GM712
          No Archiving of SCL's - use <PF2> in CC050 for Removal
 ISR....: 22________     Status: 01 - Downloaded to Development
              Archiving will be completed for ALL selected Environments
  Lib Library                                                      ed By   Date
      ESPDEMO                                                      12     20220914
      ESPDEMO   Are you sure you want to ARCHIVE these Objects     12     20220914
      ESPDEMO                                                      12     20220914
      ESPDEMO    Please confirm : Y (Y)es or (N)o                  12     20220914
      ESPDEMO                                                      12     20220914
      ESPDEMO                                                      12     20220914
      ESPDEMO  DEMO08     R  Program            15   1   Y  GM712    20220914
      ESPDEMO  DEMO09     R  Program            15   1   Y  GM712    20220914
      ESPDEMO  DEMO10     R  Program                   Y  GM712    20220914
      ESPDEMO  DEMO11     R  Program            21   2   Y  GM712    20220914
X     ESPDEMO  DEMO12     R  Program            22   1   Y  GM712    20220914
X     ESPDEMO  DEMO13     R  Program            22   1   Y  GM712    20220914
  L=Lib                      ** Start of Data **
   Restart at Library: ________ Object: ________       No of Objects: 2
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit  Arch        Scan        PgDn        Curr
```

Archive confirmation Y or N

Archive with PF4

The selected objects are removed from the EspControl object inventory including source code or object code in the selected environment.

The ISR history functions can be used as audit trail for archived objects.

The source code and object code for the modules are saved to the “fuser-backup” directory defined in the CC001 Control Variable function.

The ISR is automatically completed after the archive process terminates.

If the code does not exist in the chosen environment i.e. Allow-Source was = M and so code was removed, there will be no removal and obviously no backup as nothing exists. The object will however be removed from the Inventory List, but the ISR will not abort if code is not available for backup.

See below window for archiving of objects with examples of both backup and no backup –

```
gm712@CRONUS01:~
Executing ...
ISP250                  *** Cronus Consulting - DEV ***              09/19 18:10
ISM25001                  Select Objects for Archiving               GM712

 ISR....: 22            Status: 01 - Downloaded to Development
 Environment : Production
 ------------------------------------------------------------------------------
 Library   Object    RetCode Comments




 ESPDEMO   DEMO12            Backup of Source Code Successful
 ESPDEMO   DEMO12            Object successfully deleted
 ESPDEMO   DEMO12       3    No BACKUP taken due to NO Source Code
 ESPDEMO   DEMO12            Source Code does not exist so was not deleted
 ESPDEMO   DEMO12       3    No BACKUP taken due to NO Source Code
 ESPDEMO   DEMO12            Source Code does not exist so was not deleted

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit  Arch        Scan        PgDn        Curr
```

Archive details while processing

```
gm712@CRONUS01:~

ISP095                 *** Cronus Consulting - DEV ***                09/19 18:15
ISM095                       Display ISR History                         GM712

ISR....: 22________      Status: 99 - Completed
Xref...:                         Toggle LEFT and RIGHT for Complete Description
  Date       Time    User ID  Program  Description
20220919 18:08:58 GM712    ISP035   Added ISR Request - PRD2DEV
20220919 18:09:23 GM712    ISP035   Changed ISR Request - PRD2DEV
20220919 18:13:40 GM712    ISP250   Object: ESPDEMO DEMO12 Selected for Archiv
20220919 18:13:40 GM712    ISP250   Object: ESPDEMO DEMO13 Selected for Archiv
20220919 18:14:45 GM712    ISP250   Archiving: ESPDEMO DEMO12(New) in Developm
20220919 18:14:48 GM712    CCN250   ESPDEMO DEMO12 - Backup Taken in Developme
20220919 18:14:49 GM712    ISP250   Object successfully deleted
20220919 18:14:49 GM712    ISP250   Archiving: ESPDEMO DEMO12(New) in Test
20220919 18:14:50 GM712    CCN250   ESPDEMO DEMO12 - NO Backup in Test as no c
20220919 18:14:52 GM712    ISP250   Source Code does not exist so was not dele
20220919 18:14:52 GM712    ISP250   Archiving: ESPDEMO DEMO12(New) in Producti
20220919 18:14:53 GM712    CCN250   ESPDEMO DEMO12 - NO Backup in Production a
                          ** Start of Data **
Restart at Date: _______ Time: ____
Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Print       Quit                          PgDn        Left  Right
```

Archive ISR History Details

3.3 ISR Maintenance Function Overview (CC502)

3.3.1 CC051 – Transfer an Object to another ISR

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer

ISP051                 *** Cronus Consulting - DEV ***                09/20 11:32
ISM051                        Transfer an ISR                         GM712
 ISR....: 20_______     Status: 00 - Object Selection in Progress
 Transfer to ISR: 25_____________ * Only Status 00 or 01 can be transferred *
 Profile: PRD2DEV       Desc   : PROD TO DEV
      LIBRARY  OBJ NAME OBJ TYPE   Mode  Current ISR Stat New Saved By   Date
X     ESPDEMO  DEMO02   Program     R            20   1      GM712    20220914
X     ESPDEMO  DEMO03   Program     R            20  99      GM712    20220914

  L=Lib                        *** End of Data ***
   Restart at Library: ________ Object: ________        No of Objects: 2
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit  X-FER
```

Object Transfer between ISR's

Objects, while in the development status or any initial environment as determined by the profile linked to the ISR, may be **transferred** from one **ISR** to another. Both the **From ISR Nr** and the **Transfer To ISR Nr** must be in a development status, status 00 or 01.

This means that the objects must have either only been linked, or been downloaded **to** the "**Initial**" environment. All objects linked to the From ISR, which are in the correct status, will be reflected. All objects that need to be transferred must be selected with 'X' in the first column (see above).

Confirm by pressing <**PF4**>, and the selected objects will be transferred, with their current object status, to the new **Transfer To ISR.** This action removes these objects from their current ISR and places them in the Transfer ISR. **History** (CC095) is available on both the From and To ISR numbers.

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_eco/transfer

ISP050                  *** Cronus Consulting - DEV ***                 09/20 11:32
ISM050                    Select Objects for Transfer                   GM712

 ISR....: 25_________     Status: 00 - Object Selection in Progress
 Profile: PRD2DEV         Desc  : PROD TO DEV
 Lib Dep Library Object Mode Object Type Current ISR Stat New Saved By   Date
X _   _ ESPDEMO  DEMO02   R  Program            25   1      ESPCNTRL 20220920
X _   _ ESPDEMO  DEMO03   R  Program            25  99      ESPCNTRL 20220919
_ _   _ ESPDEMO  DEMO04   R  Program                     Y  ESPCNTRL 20220919
_ _   _ ESPDEMO  DEMO05   R  Program                     Y  GM712    20220919
_ _   _ ESPDEMO  DEMO07   R  Program                     Y  GM712    20220914
  _     ESPDEMO  DEMO08   R  Program            15   1   Y  ESPCNTRL 20220914
        ESPDEMO  DEMO09   R  Program            15   1   Y  GM712    20220914
_ _   _ ESPDEMO  DEMO10   R  Program                     Y  GM712    20220914
_ _   _ ESPDEMO  DEMO11   R  Program                        20220914
_ _   _ ESPDEMO  DEMO14   R  Program                     Y  GM712    20220914
_ _   _ ESPDEMO  NOMAP    S  Map                            ESPCNTRL 20220914
_ _   _ ESPDEMO  TST2     R  Program                        ESPCNTRL 20220912
  L=Lib                      ** Start of Data **
   Restart at Library: _______ Object: ________       No of Objects: 2
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Dupl  AddCC Quit  Error ScanD Scan        PgDn        Curr
```

Example of the Objects being transferred between ISR 20 and 25

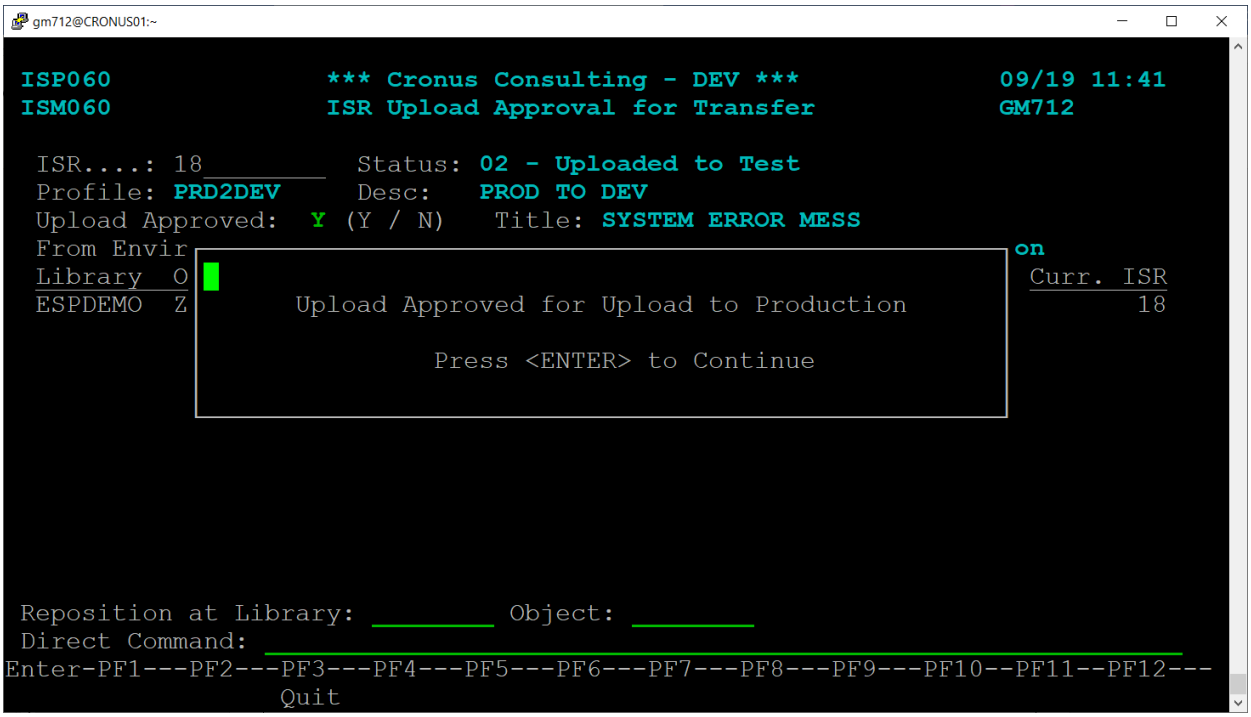
Object selection or removal may continue as normal for both ISR's 20 and 25.

The ISR transfer function is just a tool for the assistance of removing objects from one ISR and then re-selecting them to another ISR via CC050.

3.3.2 CC060 – Upload Approval ISR for Transfer

This function is used to **Upload Approve** an ISR and to set the approval status of the ISR so that CC100 can be executed for the next migration. Upload Approval cannot be set if the Return Approval is already set. This approval request is set on the linked profile via CC002 when marking A-A indicator with a “Y” or “N”. The indicator will reflect against the environment being migrated **TO**. If no approval is required (N), as specified by the profile, then CC100 can continue as normal and will return no error.

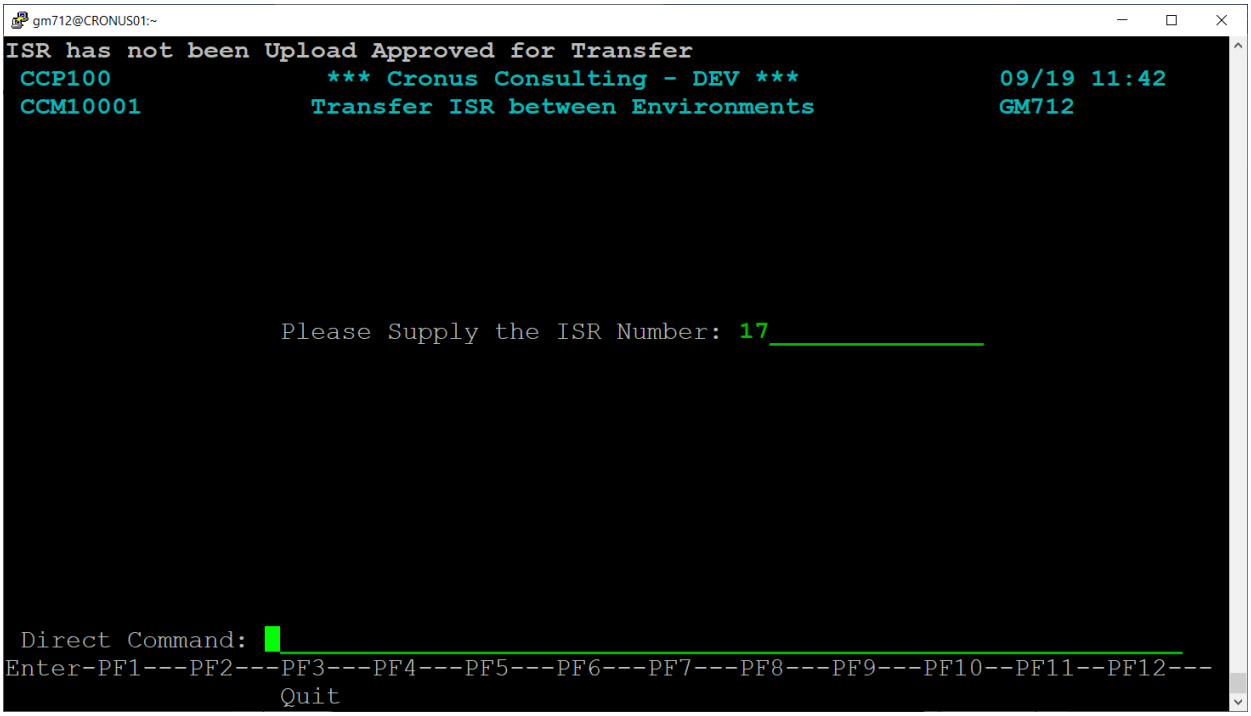
If for example, approval is required for the move to Production, then the indicator must be set to Y against the environment that will be moved to. Therefore, if Dev, Test and Prod environments exist, Production must be marked with “Y”.



Upload Approval per ISR

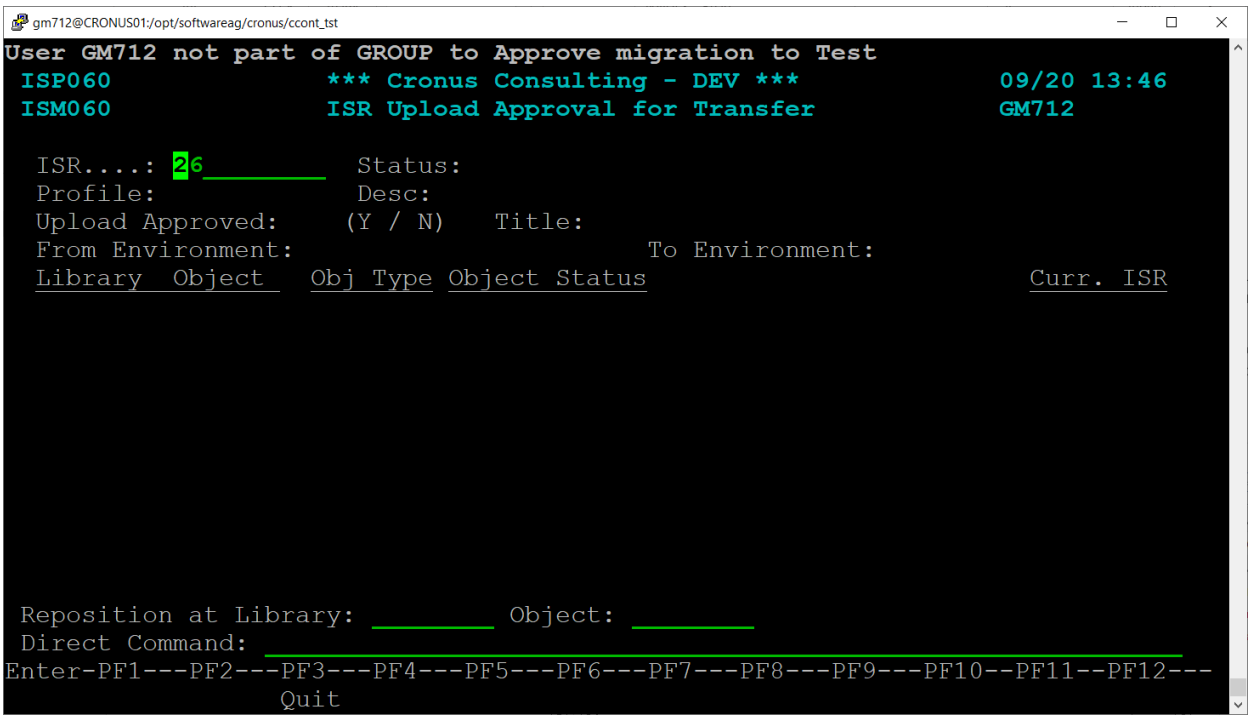
The **Upload Approval Indicator** field on Profile set-up function **CC002** is used to control the “Upload Approval” of ISR’s before they are migrated **to** the next environment. The **Upload Approval Indicator** can be set to “**Y**” or “**N**” on each of the environments. Function **CC060** – Upload Approve ISR for transfer is required for any environment where the **Upload Approval Indicator** is set. During the move from the Master to the Initial Environment (First move), no upload approval will ever be requested.

If the ISR Code SAMEAPPROV has been set to “N”, then a user other than the creator (CC030) of the ISR needs to do the Upload Approval. If the same user executes this function, an error message will be returned.



CC100 showing error where approval is required.

CC060 also checks the profile set up in CC002 for a group. If a group (not user) is linked to the environment being migrated from, the user doing the approval in CC060 must be part of this group.



CC060 approval error as user not part of the group in the linked profile

If the code APPROVE in CC010 has been amended to include USERID's other than the default APPUSER, then only this list will be allowed to approve an ISR on CC060.

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_tst
User GM712 not on Approval list to Approve migration to Test
 ISP060              *** Cronus Consulting - DEV ***            09/20 13:52
 ISM060              ISR Upload Approval for Transfer           GM712

  ISR....: 26________   Status:
  Profile:              Desc:
  Upload Approved:    (Y / N)   Title:
  From Environment:                       To Environment:
  Library  Object   Obj Type Object Status                        Curr. ISR




 Reposition at Library: ________ Object: ________
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC060 approval error as user not part of the list in CC010 on code APPROVE

3.3.3 CC070 – Mark an ISR as Completed

This function is used to change the status of an ISR to complete. Once an ISR's status has been changed to completed, all objects linked to that ISR will once again be available for selection and no more migration may take place against the completed ISR.

```
gm712@CRONUS01:~
<ENTER> to complete ISR
 ISP070                *** Cronus Consulting - DEV ***            09/19 13:10
 ISM07001                  Mark an ISR as Completed               GM712

   ISR Number: 19_____________
   ISR Status: 03 - Uploaded to Production
-------------------------------------------------------------------------------
   Profile:    PRD2DEV    PROD TO DEV
-------------------------------------------------------------------------------
 ** If Manual Complete and objects should not be restored, Mark with N **
            Restore.........: Y (Y/N/P - Yes/No/Previous ISR)
-------------------------------------------------------------------------------


           Actual Work Hours: ___1.00

            Completion Date.: 20220919



 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Complete an ISR

The following information must be entered:

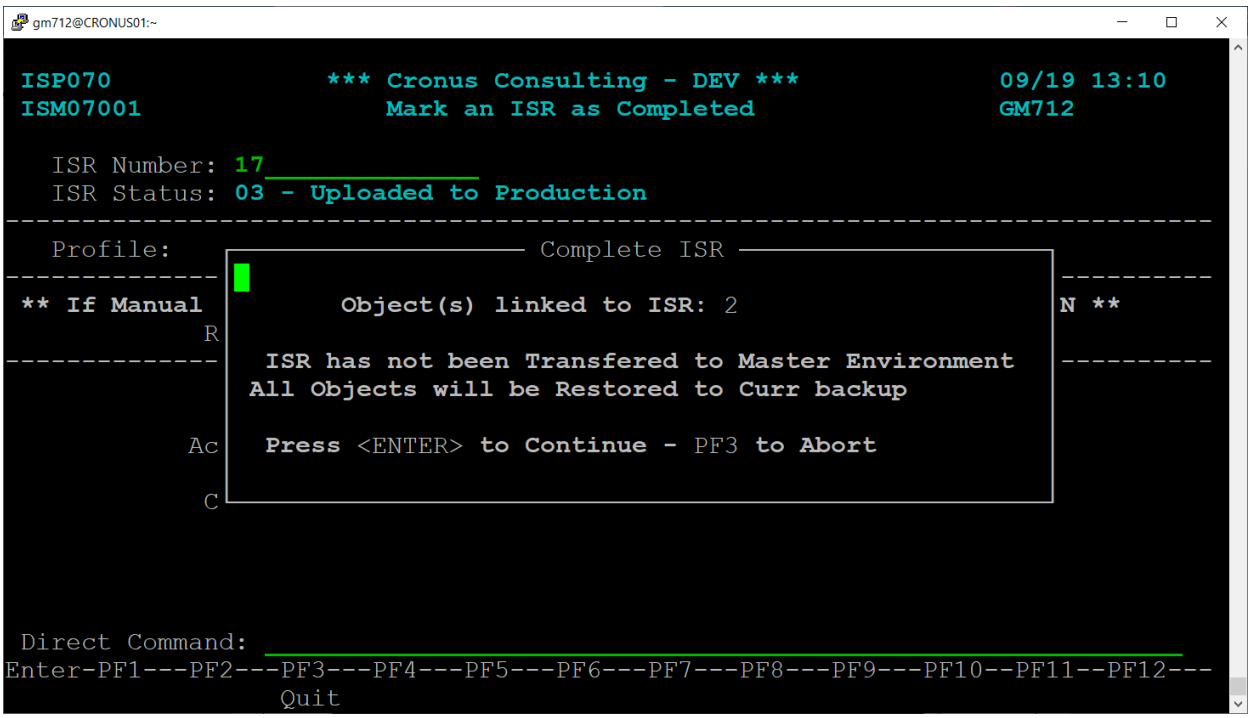
- Actual Man Hours - Number of hours used to complete the request
- Completion Date - Set to System Date
- Restore - Y/N/P – Yes, No, Previous ISR

On completion of the ISR during a CC100 migration will occur when the modules reach the Master Environment Index of the profile used. The status will be updated to 99 and all modules will be unlinked from the ISR. However, CC070 may be manually selected to end the cycle of a current ISR. The cycle will be completed and the ISR status will also then be updated to 99.

RESTORE (Y,N or P) – this will only be applicable if the ISR has not been transferred to the master environment and called from the final migration in CC100. Therefore, an early completion (manual selection of CC070) of an ISR will use this indicator.

To abort completion of the ISR press 'PF3', (by pressing <ENTER> all changes will be lost and ISR will be restored to all applicable environments, depending on the RESTORE indicator). PF3 will allow an amendment of the RESTORE indicator.

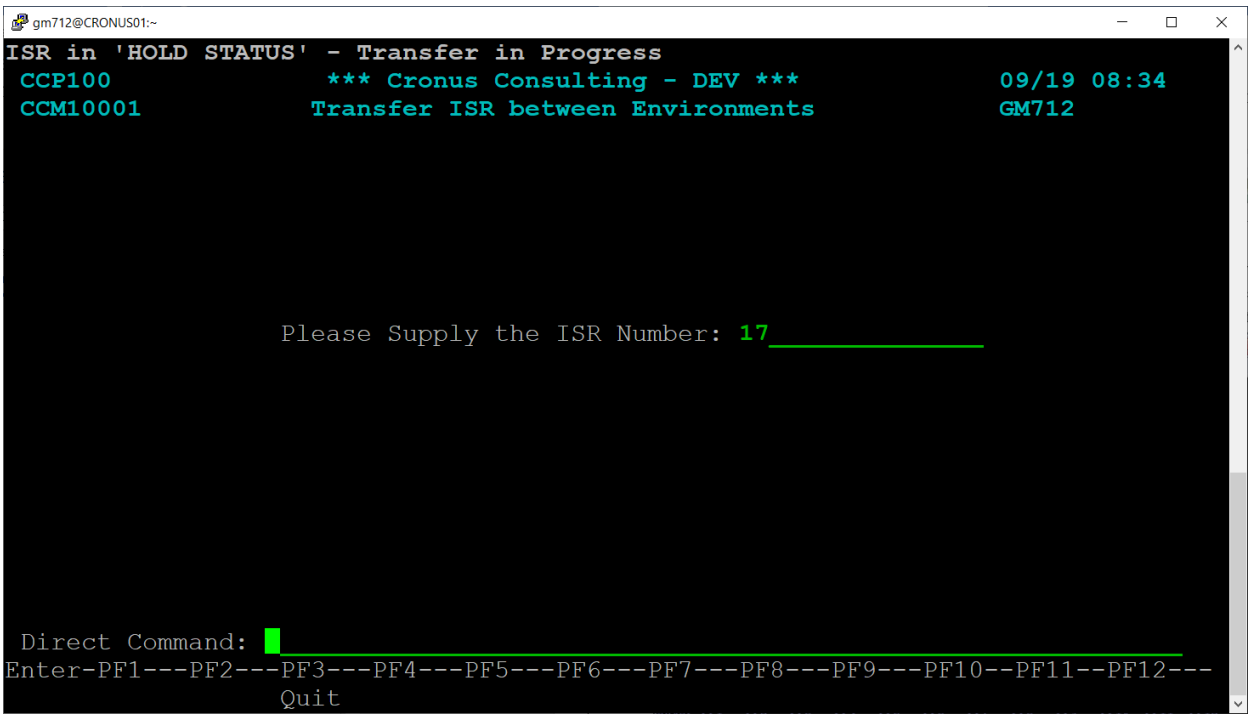
RESTORE – Y,N or P, If marked with **Y** or **P**, then objects will be restored. **Y** will use the current backup of the ISR (see backup cycle to get more detail on current backup). P will use the backup from the Previous ISR, which actually refers to the previous code before the current ISR was initiated. If marked with **N**, then ISR will be closed off as complete and the object source will be left as is, in whatever environment it has been migrated too.



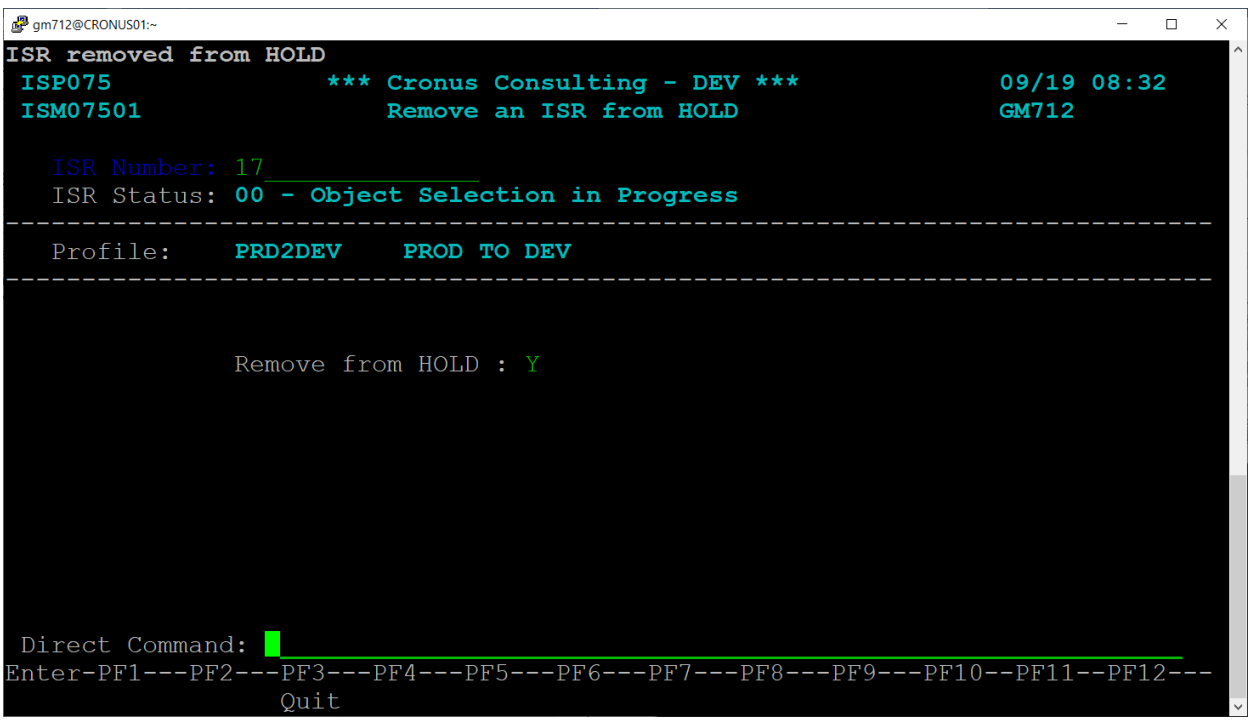
Completion Restore if NO Restore marked as N

3.3.4 CC075 – Reset ISR Status

The function is used to **reset** the status of an ISR. This should only be used if a technical problem occurred during the migration of objects or if the ISR is STOPPED via the user during a migration process due to OBJECT COLLISION or if the ISR fails the PATH VALIDATION at the start of each migration step.



Example of ISR in HOLD error



Remove an ISR from Hold

Whenever an ISR gets selected for a particular function, e.g. Linking of Objects, Migration, Return of Objects etc., the selected ISR gets automatically updated to a HOLD status. This is a security measure that prevents other users from using the same ISR when it is being worked on, or from running a function while a selected function is still busy. For example, while an ISR is busy being migrated via CC100, the ISR cannot be amended via CC050.

This HOLD indicator may be manually RESET by using the above function. This should only be done if the ISR is in HOLD due to a line error or a function did not complete correctly as specified in the manual which will force the ISR to HOLD.

Enter ISR NO and mark Remove from HOLD with a 'Y'. This will then allow the ISR to be used again.

This function should not be allocated to all users, so that ISR's do not get reset out of turn.

3.3.5 CC076 – Remove ISR from SYSOBJH Error

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_tst/transfer
ISR SYSOBJH Error reset
 ISP076              *** Cronus Consulting - DEV ***            09/20 15:22
 ISM07601            Remove an ISR from SYSOBJH Error           GM712

   ISR Number: 29
   ISR Status: 02 - Uploaded to Test
--------------------------------------------------------------------------------
   Profile:    PRD2DEV    PROD TO DEV
--------------------------------------------------------------------------------


              Reset SYSOBJH Error:  Y









 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Reset an ISR from SYSOBJ ERROR

If an ISR has been marked for Source Unload and during the migration via CC100, the Unload function did not complete successfully, then the ISR is marked as being in SYSOBJH ERROR. This allows the user to redo the migration by re-running CC100 without the objects being migrated again. This SYSOBJH ERROR indicator allows the re-migration of the ISR until the source unload has completed successfully.

If it is decided that the full migration, including the objects, must be re-executed, then this SYSERR ERROR may be RESET in CC077 as indicated above.

For security purposes, this ISR will be blocked from any other function until the source has been unloaded successfully or the SYSOBJH Error indicator has been reset.

3.3.6 CC077 – Remove an ISR from System Message Error

```
gm712@CRONUS01:/opt/softwareag/cronus/ccont_tst/transfer
ISR SYSTEM Error reset
 ISP077              *** Cronus Consulting - DEV ***            09/20 15:17
 ISM07701        Remove an ISR from System Message Error        GM712

   ISR Number: 28
   ISR Status: 00 - Object Selection in Progress
--------------------------------------------------------------------------------
   Profile:    PRD2DEV    PROD TO DEV
--------------------------------------------------------------------------------

              Reset System Message Error:  Y









 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Reset an ISR from System Message Error

If a range of Error numbers have been linked to an ISR in CC050 for the migration of the SYSERR error messages and during the migration via CC100, the SYSERR transfer function did not complete successfully, then the ISR is marked as being in SYSERR ERROR. This allows the user to redo the migration by re-running CC100 without the objects being migrated again. This SYSERR ERROR indicator allows the re-migration of the ISR until the system error messages have been transferred.

If it is decided that the full migration, including the objects, must be re-executed, then this SYSERR ERROR may be RESET in CC077 as indicated above.

For security purposes, this ISR will be blocked from any other function until the System Error Messages have been transferred successfully or the SYSERR Error indicator has been reset.

3.4 Display ISR Information Function Overview (CC503)

3.4.1 CC080 – Display ISR Status

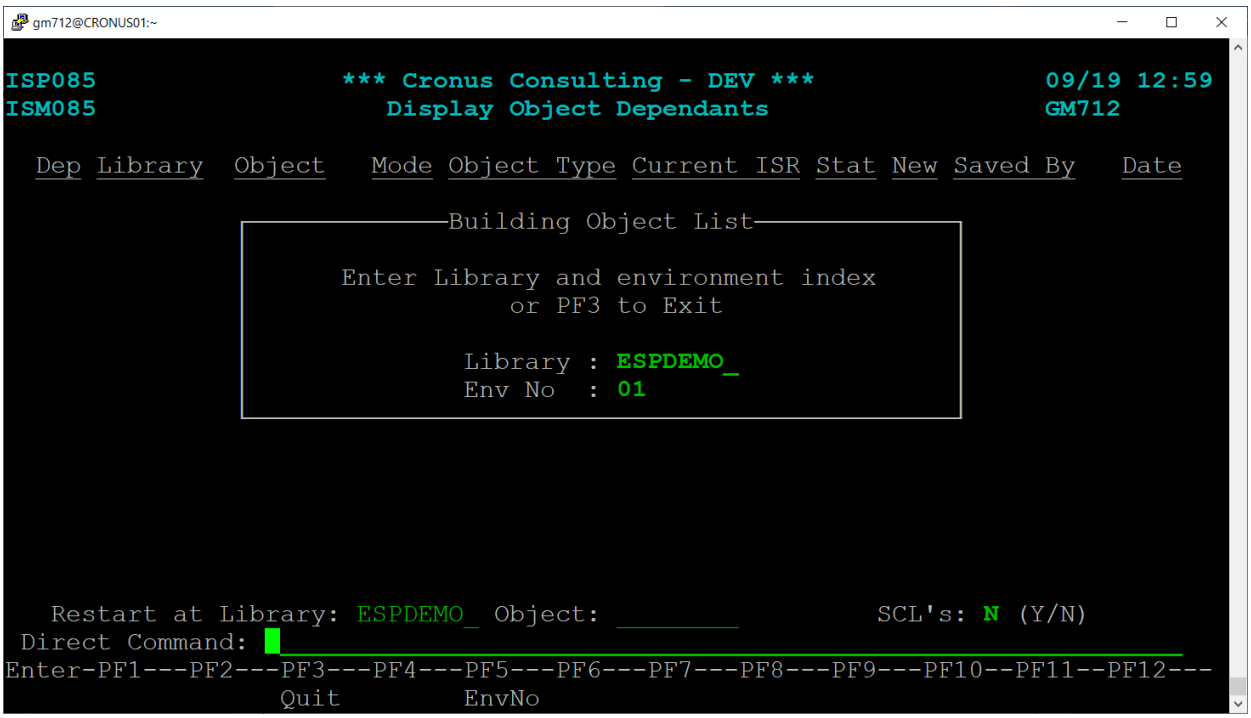
```
gm712@CRONUS01:~

ISP080              *** Cronus Consulting - DEV ***            09/19 12:52
ISM080                     Display ISR Status                  GM712

       ISR Number: 19                 Profile: PRD2DEV   PROD TO DEV
 ---------------------------------------------------------------------------
       ISR Status: 02 - Uploaded to Test
       ISR Title.: COPY FUNCTION

 System..........: ACC Access Control
                                          Date Originated.....: 20220919
 Originator......: GM712                  Allocation Date.....: 20220919
 Allocated to....: GM712                  Return Error........: N
 Telephone Ext...:                        SYSOBJH-Error.......: N
                                          System Message Error: N
 Description.....: copy function

 Date IT Received: 20220919               Estimated Work Hours: 1.00
 Date Completed..:                        Actual Work Hours...:
 Date Required...: 20220919               Rejected (Y)es/(N)o.: N

 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        Incom
```

ISR Status Enquiry

The function is used to display general information about the ISR:

3.4.2 CC085 – Display Object Dependents

The function is used to display the dependants of a selected Object:



Object Dependents Environment Number and Library Input

This function does server validation in the background and if CC001 is populated with more than one Development environment with different fusers on the Local Server, it will first bring up a window (see above screen) for the user to capture the environment number and the library. It will use this environment number to set up the correct fuser which in turn will find all the dependents of the selected object. The library will be validated against the entered environment number. This library and environment number would tie up to the initial scan in CC300 or any new objects scanned in CC050. It will use these entered parameters to determine the Inventory List for that particular environment and only display these objects. PF5 must be used to change the environment number for a new library if required. However, if the background validation determines that CC001 does not have multiple Development environments, this window will be bypassed and the full Inventory List displayed.

```
gm712@CRONUS01:~

ISP085                 *** Cronus Consulting - DEV ***                 09/19 12:53
ISM085                    Display Object Dependants                    GM712
                           Development
  Dep Library  Object   Mode Object Type Current ISR Stat New Saved By   Date
   _  ESPDEMO  DEMO01     R  Program              15   1      ESPCNTRL 20220914
   _  ESPDEMO  DEMO02     R  Program              19   2   Y  ESPCNTRL 20220919
   _  ESPDEMO  DEMO03     R  Program              19   2   Y  ESPCNTRL 20220919
   _  ESPDEMO  DEMO04     R  Program              17   2   Y  ESPCNTRL 20220919
   _  ESPDEMO  DEMO05     R  Program                       Y  GM712    20220914
   _  ESPDEMO  DEMO07     R  Program                       Y  GM712    20220914
   _  ESPDEMO  DEMO08     R  Program              15   1   Y  ESPCNTRL 20220914
   _  ESPDEMO  DEMO09     R  Program              15   1   Y  GM712    20220914
   _  ESPDEMO  DEMO10     R  Program                       Y  GM712    20220914
   _  ESPDEMO  DEMO11     R  Program                       Y  GM712    20220914
   _  ESPDEMO  DEMO12     R  Program                       Y  GM712    20220914
   _  ESPDEMO  DEMO13     R  Program                       Y  GM712    20220914
   _  ESPDEMO  DEMO14     R  Program                       Y  GM712    20220914
   _  ESPDEMO  NOMAP      S  Map                              ESPCNTRL 20220914
                             ** Start of Data **
   Restart at Library: _______ Object: ________        SCL's: N (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                          PgDn
```

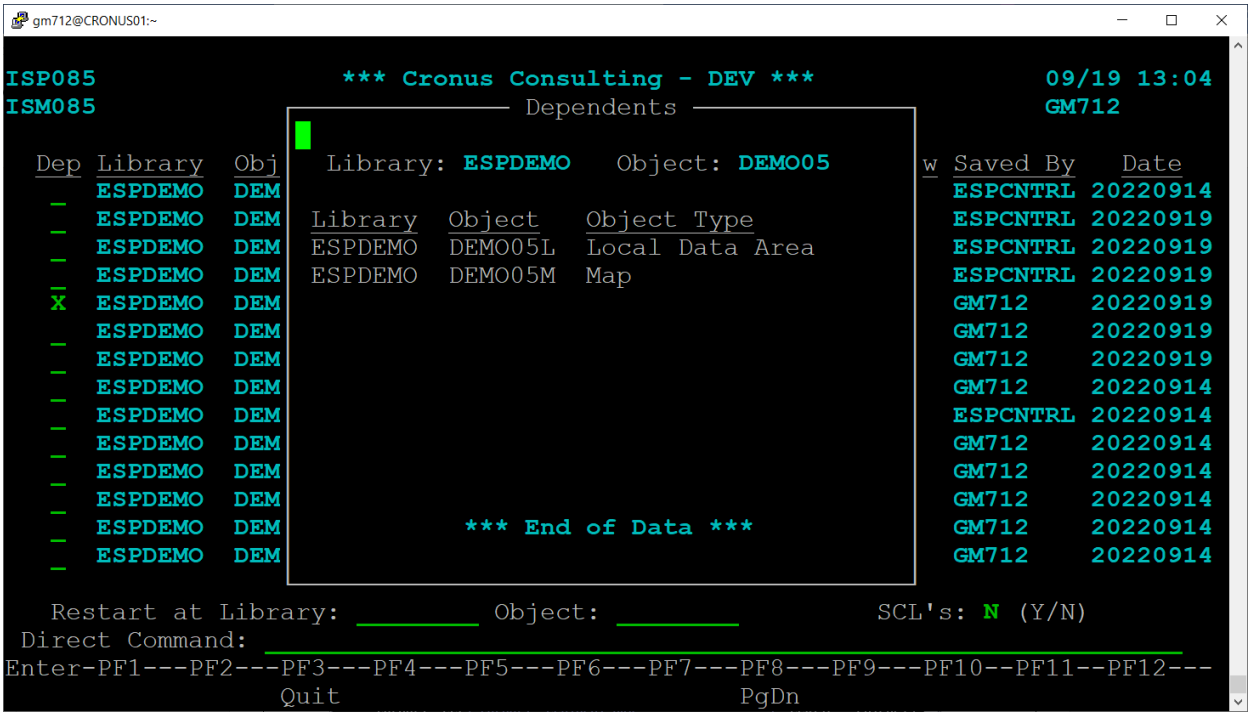
Object Dependents

Information regarding the library and its objects are displayed:

- Library Name - Object Library Name (Scanned in Library)
- Object - Object Name
- Mode - Structured or Reporting
- Object Type - Object Type as defined in Natural
- Current ISR - ISR number - Blank if not linked
- Object Status - Status of object
- Object new indicator - New or old indicator
- Saved information - User that last saved the object
- Date Saved - Date last saved

Select Objects with ‘X’

A list of the dependants found for the object is displayed detailing the **Object Name** and **Object Type.**



Dependents Displayed for Selected Object

The object dependants detail:

- Library Name - Library Name of Object
- Object Name - Object Name
- Object Type - Description of object type

Only 10 objects are displayed per page – use **PF8** to page forward and **PF7** to page back.

NOTE : This function will not give the correct results if the code has been “moved” from or only “object code” exists in the scanned environment, it will only reflect objects where source is currently residing in the Development Areas, as this function reads through the actual code that exists in the fuser.

3.4.3 CC088 – Display Object History

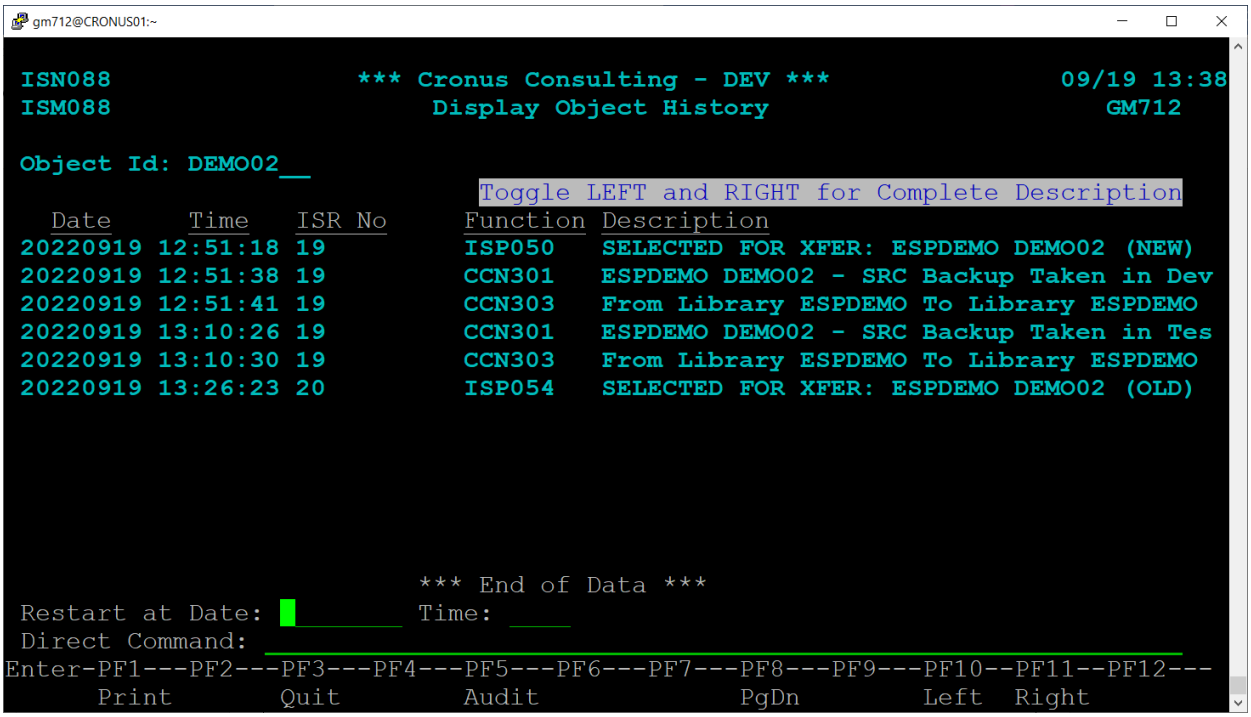
A list of all objects in the Object Inventory (scanned objects) will be displayed, and the user can choose as many objects, by marking with an 'X', as he likes. This will in turn display all the history pertaining to this object using the CC095 function.

```
gm712@CRONUS01:~

ISP088                *** Cronus Consulting - DEV ***            09/19 13:37
ISM08801                   Display Object History                GM712
                                          Current
  Library   Object    Object Type   Stat    ISR
_ ESPDEMO   DEMO01    PROGRAM        01        15
X ESPDEMO   DEMO02    PROGRAM        99        20
_ ESPDEMO   DEMO03    PROGRAM        99        20
_ ESPDEMO   DEMO04    PROGRAM
_ ESPDEMO   DEMO05    PROGRAM
_ ESPDEMO   DEMO07    PROGRAM
_ ESPDEMO   DEMO08    PROGRAM        01        15
_ ESPDEMO   DEMO09    PROGRAM        01        15
_ ESPDEMO   DEMO10    PROGRAM
_ ESPDEMO   DEMO11    PROGRAM        02        21
_ ESPDEMO   DEMO12    PROGRAM
_ ESPDEMO   DEMO13    PROGRAM
_ ESPDEMO   DEMO14    PROGRAM
_ ESPDEMO   NOMAP     MAP

  Restart at Library: ________ Object: ________          SCL's: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit              Old    PgUp   PgDn
```

Object List for History Selection

The information displayed includes:

- 'X' (to select) - Selected Object(s) to be for History Display
- Library Name - Library Name of Object
- Object Name - Object Name
- Object Type - Object Type
- Object Status - Object status, which equals the status of the selected profile's environment where the object has currently been migrated to. 99 is for Complete.
- Current ISR Number - Linked ISR number (if still linked to an ISR)



ISR History per Object chosen

The above screen is the result of marking ‘X’ in the selection field and displays all History information (as in CC095 when an ISR is selected), per ISR. This window, by using PF8 to page down and PF7 to page up, will reflect **ALL** ISR’s that the selected object was linked to and all the relevant information pertaining to that object in that ISR.

Use PF11 to scroll to the right and PF10 and scroll to the left to see more information per screen.

Display History for Archive Objects using PF6

```
gm712@CRONUS01:~

ISP088                *** Cronus Consulting - DEV ***              09/19 18:16
ISM08801                   Display Object History                  GM712
                                          Current
  Library   Object    Object Type   Stat     ISR
_ ESPDEMO   DEMO01    PROGRAM        01        15
_ ESPDEMO   DEM ──────────────────Archived Objects──────────────
_ ESPDEMO   DEM
_ ESPDEMO   DEM  Enter Archived Object ID to show all History
_ ESPDEMO   DEM                or PF3 to Exit
_ ESPDEMO   DEM          Object Id: DEMO12__
_ ESPDEMO   DEM
_ ESPDEMO   DEM
_ ESPDEMO   DEM
_ ESPDEMO   DEMO11    PROGRAM        02        21
_ ESPDEMO   DEMO14    PROGRAM
_ ESPDEMO   NOMAP     MAP
_ ESPDEMO   TST2      PROGRAM
_ ESPDEMO   TST3      PROGRAM

  Restart at Library: ________ Object: ________          SCL's: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                Old   PgUp  PgDn
```

History for Archived Objects

```
gm712@CRONUS01:~

ISN088                *** Cronus Consulting - DEV ***              09/19 18:17
ISM088                     Display Object History                     GM712

Object Id: DEMO12__
                          Toggle LEFT and RIGHT for Complete Description
  Date       Time     ISR No      Function Description
20220919 18:13:40 22              ISP250   Object: ESPDEMO DEMO12 Selected for Arch
20220919 18:14:45 22              ISP250   Archiving: ESPDEMO DEMO12(New) in Develo
20220919 18:14:48 22              CCN250   ESPDEMO DEMO12 - Backup Taken in Develop
20220919 18:14:49 22              ISP250   Object successfully deleted
20220919 18:14:49 22              ISP250   Archiving: ESPDEMO DEMO12(New) in Test
20220919 18:14:50 22              CCN250   ESPDEMO DEMO12 - NO Backup in Test as no
20220919 18:14:52 22              ISP250   Source Code does not exist so was not de
20220919 18:14:52 22              ISP250   Archiving: ESPDEMO DEMO12(New) in Produc
20220919 18:14:53 22              CCN250   ESPDEMO DEMO12 - NO Backup in Production
20220919 18:14:56 22              ISP250   Source Code does not exist so was not de


                          *** End of Data ***
 Restart at Date: ________ Time: ____
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Print        Quit        Audit              PgDn        Left  Right
```

Actual History Enquiry for DEMO12 as selected above

This history enquiry will provide the details of a backup if taken, and this backup could be restored via SYSMAIN Import if required. There is no formal restore utility due to the objects no longer being in CC050 Object Inventory.

3.4.4 CC090 – Display Objects Linked to an ISR

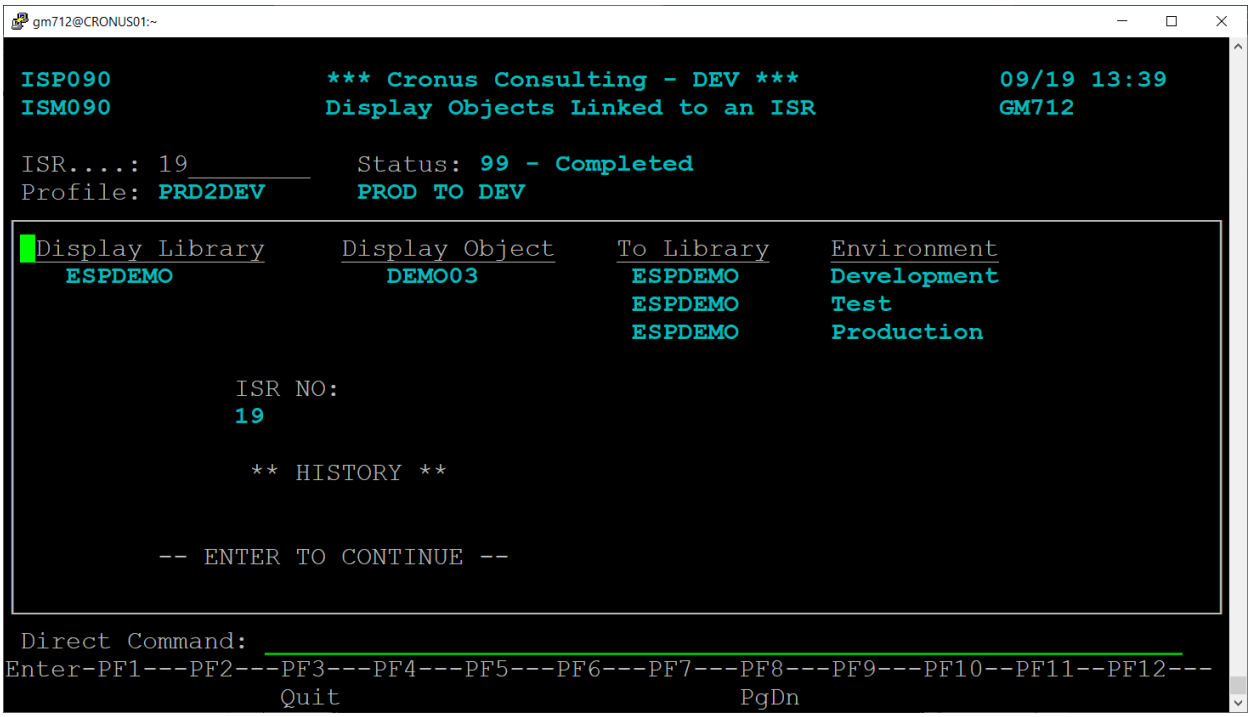
This function is used to display the status of the ISR and all the objects linked to the ISR during the change request. Once an ISR has been completed the objects will still be displayed, however the Current ISR column will either be blank or contain the ISR No to which the object is now currently linked.

```
gm712@CRONUS01:~

ISP090                *** Cronus Consulting - DEV ***            09/19 13:39
ISM090                Display Objects Linked to an ISR           GM712

ISR....: 19_________    Status: 99 - Completed
Profile: PRD2DEV        PROD TO DEV
'X' Library   Object Type Object Status                          Curr. ISR
 _  ESPDEMO   DEMO02   P  99 - Completed                               20
 _  ESPDEMO   DEMO03   P  99 - Completed                               20

                        *** End of Data ***
Reposition at Library: _________ Object: _________
Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                            PgDn
```

Objects linked per ISR

The information displayed includes:

- 'X' (to select) - Reflects the Environment and Libraries used for the selected Object during the cycle of the ISR
- Library Name - Library Name of Object
- Object Name - Object Name
- Object Type - Object Type
- Object Status - Object status, which equals the status of the selected profile's environment where the object has currently been migrated to.
  99 is for complete
- Current ISR Number - Linked ISR number (if still linked to an ISR)

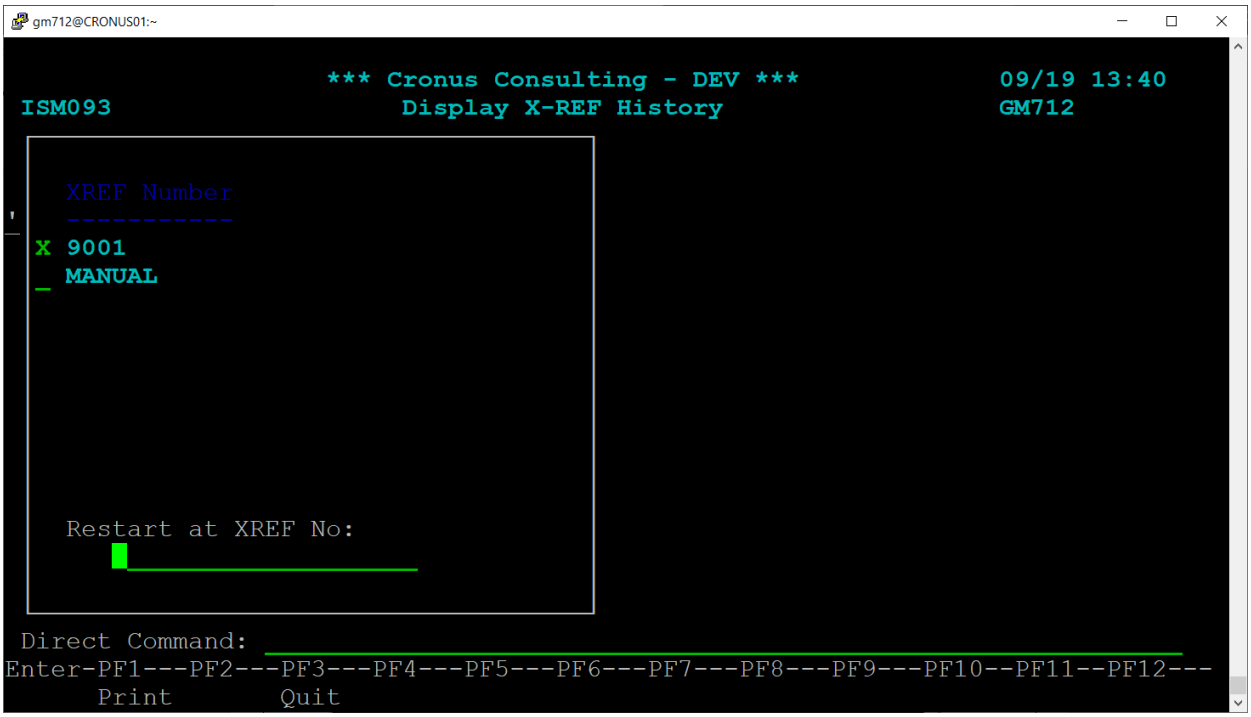


Reflects Libraries per Environment of Selected Object

The above screen is the result of marking ‘X’ in the selection field next to the object name and will show the migration path and the libraries used in each environment. This function will show the history of the library and environment movement, even if the profile has since changed for new ISR’s.

3.4.5 CC093 – Display X-REF History

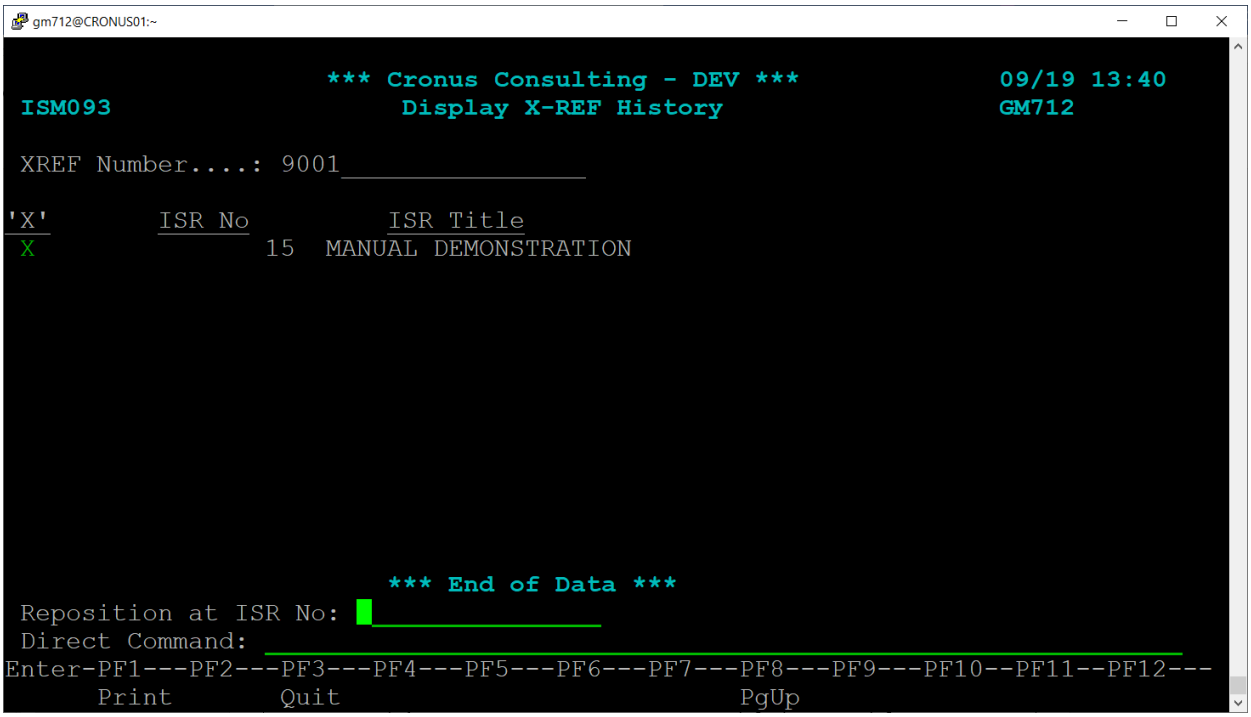
This function is used to display the History of all ISR's that have been linked to a specific X-REF Number. This X-REF Number is informational only but can be linked to a user exit ISNUX030 where site specific code can be entered to validate this X-REF Number. A list of all X-REF numbers used in migrations will be displayed, if function entered without an X-REF number, and these X-REF numbers may then be selected. This will in turn display all ISR numbers linked to this X-REF and then the history pertaining to each selected ISR. One X-REF Number may have many ISR's linked.



Help on X-REF numbers when entering in CC093

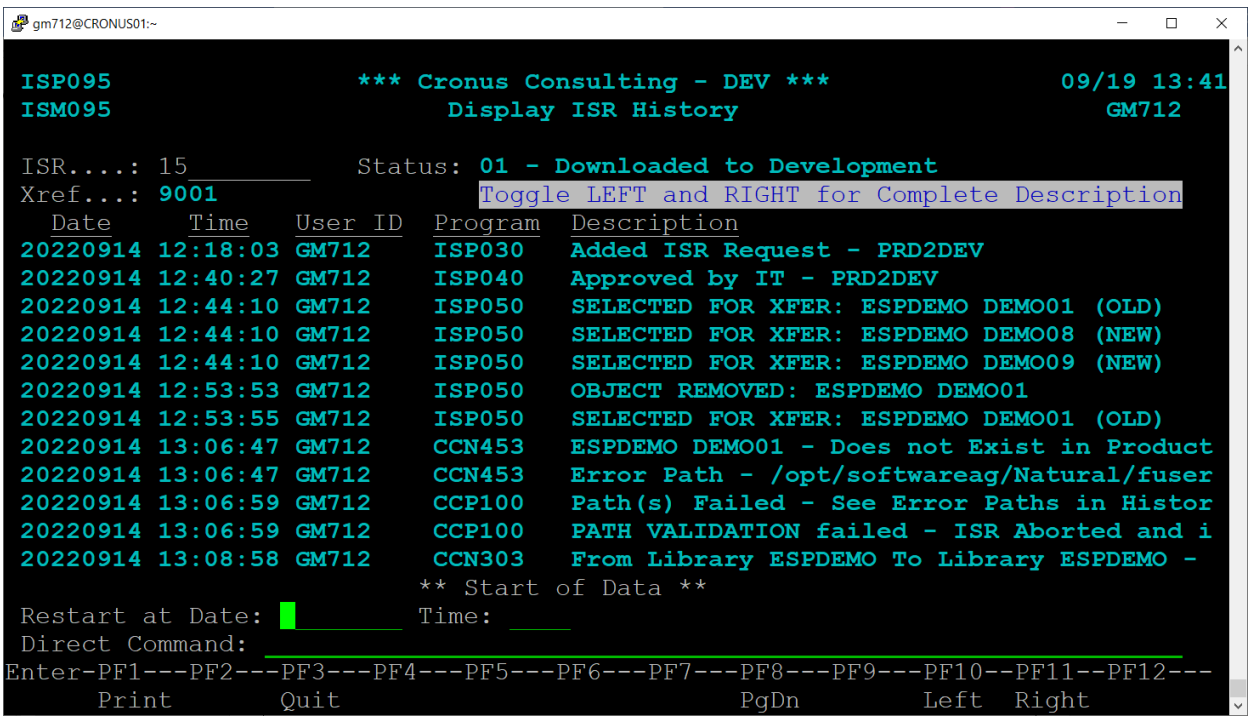
The information displayed includes:

- 'X' (to select) - Selected ISR(s) for History Display
- ISR No - Linked ISR No
- ISR Title - The title associated with the ISR set up in CC030



Display X-REF History per ISR

On selection of as many ISR's as required, the function will in turn for each selected ISR, reflect the History screen CC095. PF3 to exit each History screen to see the next selected ISR. On PF3 of the last selected ISR, the user will be returned to the X-Ref Display main window.



Display ISR History for selected XREF

3.4.6 CC095 – Display ISR History

This function provides an audit trail of the ISR's progress with date's, time's and descriptions of the various actions performed during the selection, migration, backup or restore of any particular ISR.

```
gm712@CRONUS01:~

ISP095                 *** Cronus Consulting - DEV ***                09/19 13:34
ISM095                       Display ISR History                         GM712

ISR....: 21________      Status: 02 - Uploaded to Test
Xref...:                         Toggle LEFT and RIGHT for Complete Description
  Date       Time    User ID  Program  Description
20220919 13:33:24 GM712    ISP035   Added ISR Request - MOV3PRD
20220919 13:33:37 GM712    ISP050   SELECTED FOR XFER: ESPDEMO DEMO11 (NEW)
20220919 13:34:33 GM712    CCN301   ESPDEMO DEMO11 - SRC Backup Taken in Devel
20220919 13:34:36 GM712    CCN303   From Library ESPDEMO To Library ESPDEMO -
20220919 13:34:41 GM712    CCP100   Object ESPDEMO - DEMO11 Scratched in Devel
20220919 13:34:41 GM712    CCP100   Uploaded from Development to Test - SOURCE

                          *** End of Data ***
Restart at Date: ________ Time: ____
Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Print       Quit                          PgDn        Left  Right
```

ISR History Enquiry

The information displayed includes:

- Date - Date transferred
- Time - Time of transfer
- User Id - User that executed the transfer
- Program - EspControl Program executed (function)
- Description : - Description of the actual event, including environments, paths, library and object names
- XREF Number - The XREF number captured per ISR in CC030 if Applicable
- PF5 - Audit Facility (see explanation below)

The description is more than one screen in width. Use **PF11** to toggle to the Right to see more information and **PF10** to return to the left again.

Right hand side of the History Display screen

```
gm712@CRONUS01:~

ISP095                 *** Cronus Consulting - DEV ***                09/19 13:34
                             Display ISR History                         GM712

ISR....: 21            Status: 02 - Uploaded to Test

Description
Added ISR Request - MOV3PRD
SELECTED FOR XFER: ESPDEMO DEMO11 (NEW)
ESPDEMO DEMO11 - SRC Backup Taken in Development
From Library ESPDEMO To Library ESPDEMO - DEMO11 Copied
Object ESPDEMO - DEMO11 Scratched in Development
Uploaded from Development to Test - SOURCE - STOW - BACKUP

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Print       Quit                          PgDn        Left  Right
```

ISR History Enquiry – PF11 to the right

Detailed information regarding the ISR is kept – every phase of the migration is recorded and logged on the ESP audit history file – ISR creation, ISR approval, Linking of Objects, Transfer, Return, Restores, Archiving etc.

PF5 – Audit Facility

By pressing PF5, the user will execute the audit facility. This audit is a reflection of events between a start and end date entered by the user, as well as a selection of the required environments, and will in turn produce either a report or .csv file, depending on the selection by the user. The start and end date will refer to the creation of the ISR (when created via CC030) and the end date will refer to the completion of the ISR (CC070).

The audit events reflected will be per object, using input data, as follows:

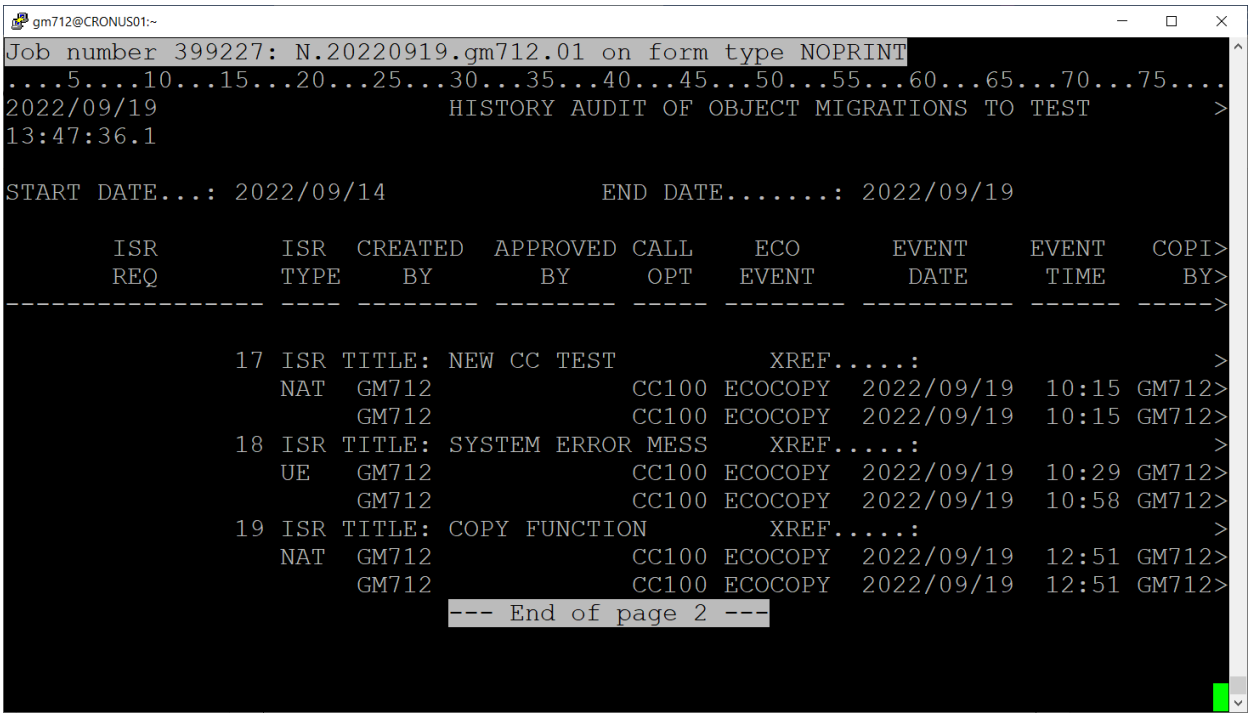
- The creation of the ISR and user, the approval of the ISR and user, the migration of the ISR and user, date and time and the function requested
- ECOCOPY – the migration of an object
- ECORETN – the return of an object
- ECOREST – the restore of an object
- ECORESTA – the restore of an object with error
- ECOARCH – Archive to an environment

```
gm712@CRONUS01:~

ISP195                *** Cronus Consulting - DEV ***              13:44:27.7
ISM195               Audit History Report or .csv File             2022/09/19

Start Date:           2022/01/01                        (Format CCYY/MM/DD)
End Date   :          2022/09/19
                      Objects: Y  or  SCL's..: N

Print Report:         N
.csv File    :        Y

     .csv file name ==> /dev shell script path/transfer/userid_date_time.csv


** ISR Completion Date will be used for the parameter Start and End Date **
** Therefore only COMPLETED ISR's will be included


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Exit
```

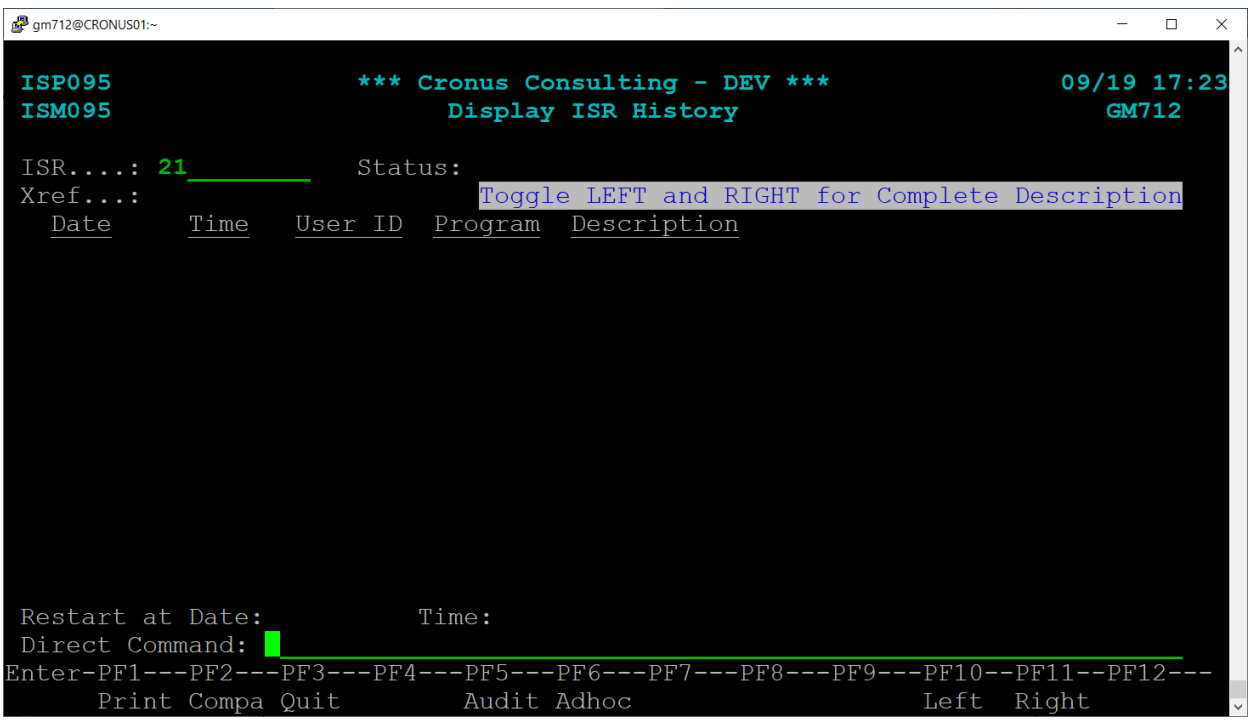
PF5 – Audit input screen

```
gm712@CRONUS01:~

ISP195                *** Cronus Consulting - DEV ***              13:45:55.6
ISM195               Audit History Report or .csv File             2022/09/19

Start Date:           2022/01/01                        (Format CCYY/MM/DD)
End Date   :          2022/09/19
                      Objects: Y  or  SCL's..: N

Print Re   .csv FILE has been created... see path below to FTP
.csv Fil
                 /opt/softwareag/cronus/ccont_eco/transfer/
     .cs          FILE NAME - AUDITGM712_20220919_1344.csv          time.csv
                            Enter to Continue

** ISR Completion Date will be used for the parameter Start and End Date **
** Therefore only COMPLETED ISR's will be included


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Exit
```

Audit .csv file Confirmation Window



Audit Report Example



PF2 Compare, PF6 Adhoc

PF2 – Compare Log

COMPARE is a function from CC200 where the source is moved from Production to Library COMPARE in Development, so that source comparisons can be done between Production and Devlopment. PF2 will show the log for the COMPARE function. It always defaults to ISR 99999999.

```
gm712@CRONUS01:~

ISP095                 *** Cronus Consulting - DEV ***                09/19 18:02
ISM095                       Display ISR History                         GM712

ISR....: 99999999__    Status: 00 - Object Selection in Progress
Xref...:                       Toggle LEFT and RIGHT for Complete Description
  Date      Time    User ID  Program  Description
20220919 18:02:25 GM712    CCN203   DEMO02 : moved to DEV Compare - CCP200
20220919 18:02:46 GM712    CCN203   DEMO03 : moved to DEV Compare - CCP200
20220919 18:02:51 GM712    CCN203   DEMO01 : moved to DEV Compare - CCP200

                           *** End of Data ***
Restart at Date: _______ Time: ____
Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Print       Quit                          PgDn        Left  Right
```

Compare Log

PF6 – Adhoc Log

ADHOC is a function from CC200 where the source is moved from a Development Adhoc Library, defined in code ADHOCL in CC010, to Production directly, without having to create an ISR. PF6 will show the log for the ADHOC function. It always defaults to ISR 77777777.

```
gm712@CRONUS01:~

ISP095                 *** Cronus Consulting - DEV ***                09/19 18:06
ISM095                       Display ISR History                         GM712

ISR....: 77777777__    Status: 00 - Object Selection in Progress
Xref...:                       Toggle LEFT and RIGHT for Complete Description
  Date      Time    User ID  Program  Description
20220919 18:04:58 GM712    CCN204   DEMO05 : moved to Prod ADHOC  - CCP200
20220919 18:05:12 GM712    CCN204   DEMO07 : moved to Prod ADHOC  - CCP200
20220919 18:05:28 GM712    CCN204   DEMO05L : moved to Prod ADHOC  - CCP200
20220919 18:06:01 GM712    CCN204   DEMO05M : moved to Prod ADHOC  - CCP200

                           *** End of Data ***
Restart at Date: _______ Time: ____
Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Print       Quit                          PgDn        Left  Right
```

Adhoc Log

3.4.7 CC097 - Display Object Status

Displays a list off all objects currently defined in the Inventory of EspControl.

```
gm712@CRONUS01:~

ISP097                 *** Cronus Consulting - DEV ***                09/19 13:48
ISM097                      Display Object Status                     GM712

   Library   Object    Mode Object Type Stat New Current ISR ISR Title
 _ ESPDEMO   DEMO01     R   PROGRAM      1               15
 _ ESPDEMO   DEMO02     R   PROGRAM     99               20
 X ESPDEMO   DEMO03     R   PROGRAM     99               20
 _ ESPDEMO   DEMO04     R   PROGRAM
 _ ESPDEMO   DEMO05     R   PROGRAM
 _ ESPDEMO   DEMO07     R   PROGRAM
 _ ESPDEMO   DEMO08     R   PROGRAM      1               15
 _ ESPDEMO   DEMO09     R   PROGRAM      1               15
 _ ESPDEMO   DEMO10     R   PROGRAM
 _ ESPDEMO   DEMO11     R   PROGRAM      2               21
 _ ESPDEMO   DEMO12     R   PROGRAM
 _ ESPDEMO   DEMO13     R   PROGRAM
 _ ESPDEMO   DEMO14     R   PROGRAM
 _ ESPDEMO   NOMAP      S   MAP

   Restart at Library: ________ Object: ________          SCL's: N (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

Current Object Status

If an object is selected with a ‘X”, a list of all ISR’s that the object has been linked to will be displayed

The following information is given per object:

- Library - Scanned in Library in the Dev environment
- Object Name - Object name
- Object Mode - Structured/Reporting mode
- Object Type - Object type e.gg Program/Map/Local
- Object Status - Object status - Blank if not linked to an ISR
- New/Old indicator - Y for New and Blank for Old
- Current ISR No - Current ISR nr. - Blank if not linked to an ISR
- ISR Title - Current ISR Title - Blank if not linked to an ISR

A list of all ISR’s that the object has been linked to will be displayed by selecting the object.

```
gm712@CRONUS01:~

ISN097                *** Cronus Consulting - DEV ***            09/19 13:50
ISM09702                  Object Status per ISR                  GM712

Library: ESPDEMO  Object: DEMO03

   ISR No ISR Title            Originator          Completed  Developer
       19 COPY FUNCTION        GM712               2022-09-19 GM712
       20 COPY FUNCTION        GM712                          GM712


                        *** End of Data ***
  Restart at ISR No: _________
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                            PgDn
```

ISR History per Object selected

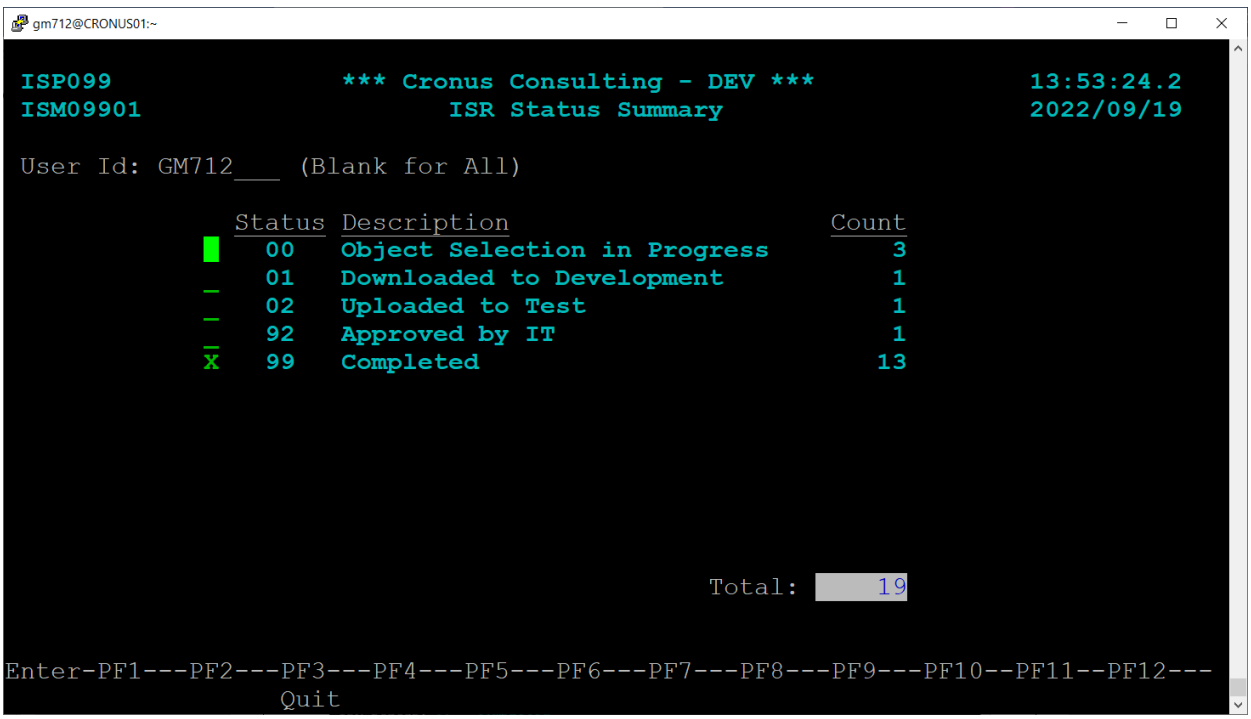
The following information history is displayed per object:

- ISR No - Previously linked ISR's
- ISR Title - Title of ISR
- Originator - Originator of the ISR
- Completed - Date ISR was completed
- Developer - The name of the developer

NOTE: This function displays a list of all objects that have been scanned into the Object Inventory List, whether old or new objects, but NOT archived objects.

3.4.8 CC099 - ISR Status Summary

This function will display a summary of all ISR's and their current statuses (status description is dependent on the linked Profile) in a summarised format that have been created by the Userid entered. The summary can also be displayed for ALL users by omitting the User-ID.



ISR Status Enquiry reflecting ALL ISR's

The following information history is displayed per user:

- Status - EspControl Status
- Description - Description of status
- Count - Number of ISR's with the associated status

To view all ISR's in a particular status – select the relevant status with an 'X'.

```
gm712@CRONUS01:~

ISP099                *** Cronus Consulting - DEV ***                13:53:07.3
ISM09902                     ISR Status Summary                      2022/09/19

ISR Status: 99 - Completed
                                                Request    Est.      Date
User ID  ISR No Title              Profile      Date      Time    Completed
GM712         3 TEST SCL           SCL2DEV    20220817     1.00   20220817  SCL
GM712         4 TEST NO COMPILE    PRD2DEV    20220818     1.00   20220818
GM712         5 TEST NO COMPILE    PRD2DEV    20220818     1.00   20220824
GM712         6 TEST NO COMPILE    PRD2DEV    20220824     1.00   20220824
GM712         7 SCL ADHOC TEST     SCL2DEV    20220826     1.00   20220910  SCL
GM712         8 TEST               PRD2DEV    20220908     1.00   20220908
GM712         9 TEST               PRD2DEV    20220909     1.00   20220909
GM712        10 TEST               PRD2DEV    20220909     1.00   20220909
GM712        13 INITIAL BACKUP     BACKUP01   20220914     1.00   20220914
GM712        14 INITIAL BACKUP     BACKUP01   20220914     1.00   20220914
GM712        17 NEW CC TEST        PRD2DEV    20220918     1.00   20220919
GM712        18 SYSTEM ERROR ME    PRD2DEV    20220919     1.00   20220919

Restart at - User: ________ Isr No: ____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

ISR Status Selection

The following information history is displayed per status as shown above:

- User ID - User-id of creator of the ISR
- ISR No - ISR Number
- ISR Title - ISR Title
- Profile - Profile Name
- Request Date - Date ISR was requested
- Est Time - Estimated man hours
- Date Completed - Date the ISR was completed

3.5 ISR RETURN AND RESTORE FUNCTION OVERVIEW (CC504)

3.5.1 CC053 – Link Objects per ISR for RETURN

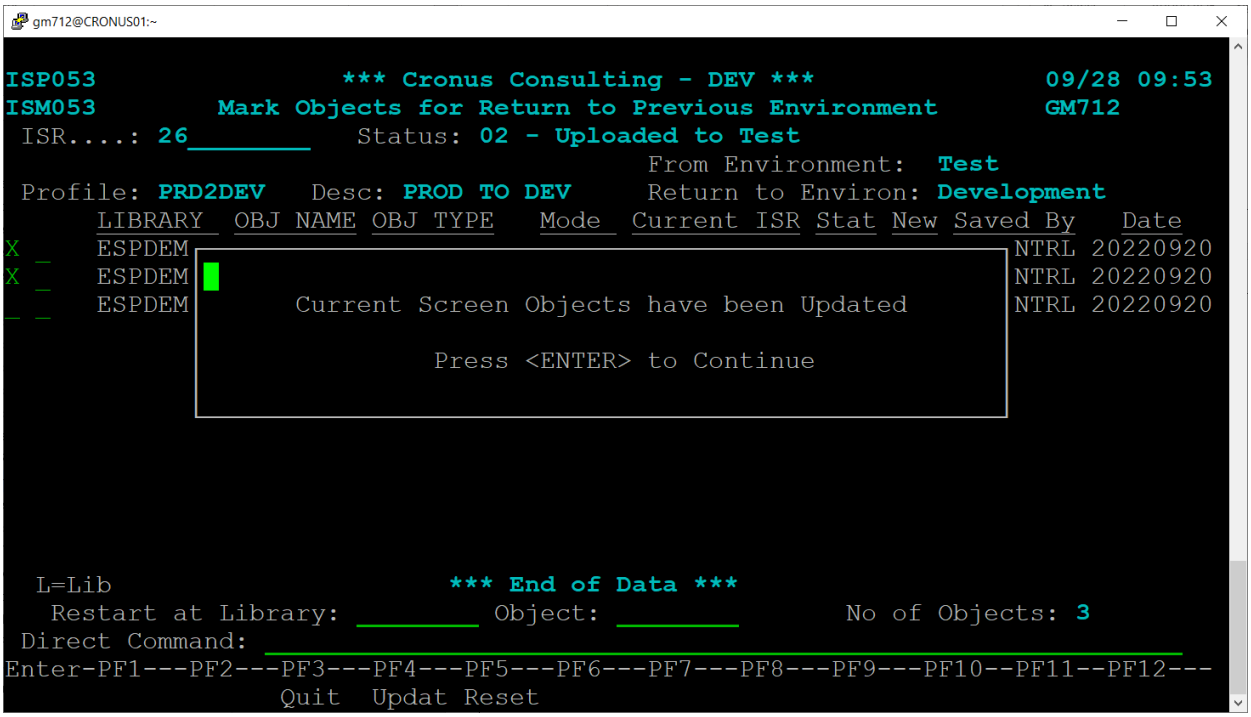
This function is used to select all objects that will form part of the requested ISR that is to be returned to the previous environment.

```
gm712@CRONUS01:~
ISP053                 *** Cronus Consulting - DEV ***                 09/28 09:53
ISM053         Mark Objects for Return to Previous Environment         GM712
 ISR....: 26________     Status: 02 - Uploaded to Test
                                         From Environment:  Test
 Profile: PRD2DEV    Desc: PROD TO DEV    Return to Environ: Development
      LIBRARY  OBJ NAME OBJ TYPE   Mode  Current ISR Stat New Saved By   Date
X _   ESPDEMO  DEMO02   Program      R           26   2      ESPCNTRL 20220920
X _   ESPDEMO  DEMO03   Program      R           26   2      ESPCNTRL 20220920
_ _   ESPDEMO  DEMO04   Program      R           26   2   Y  ESPCNTRL 20220920

  L=Lib                        *** End of Data ***
   Restart at Library: ________ Object: ________       No of Objects: 3
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit  Updat Reset
```

Link Objects to ISR for Return to Previous Environment

Once the ISR No has been entered a list of all objects already LINKED to that ISR will be displayed and then these objects must be marked for Return to a previous environment. Only the objects selected will be returned to the previous environment. Only objects already in the second and upper environment may be linked, as if the object is already in the Initial Environment, it cannot be returned anywhere. If an ISR is complete, i.e. in the Master Environment, then the ISR cannot be returned either. In this scenario, a new ISR must be opened for the migration of these objects to be downloaded to Development.

Objects can now be linked by entering an ‘X’ next to the object in the New Column. Once selection is complete for the displayed screen, press PF4 for the updating of these linked objects. A confirmation window will be displayed as reflected below. The user may now move forwards via PF8 to the next available screen and then link and update via PF4 again, until all linked objects have been viewed or move back to the previous screen via PF7 or exit the transaction using PF3. The objects must first be approved via CC061 (if the code RETAPP = Y in CC010, else no return approval is required) and then the actual return executed via function CC102.

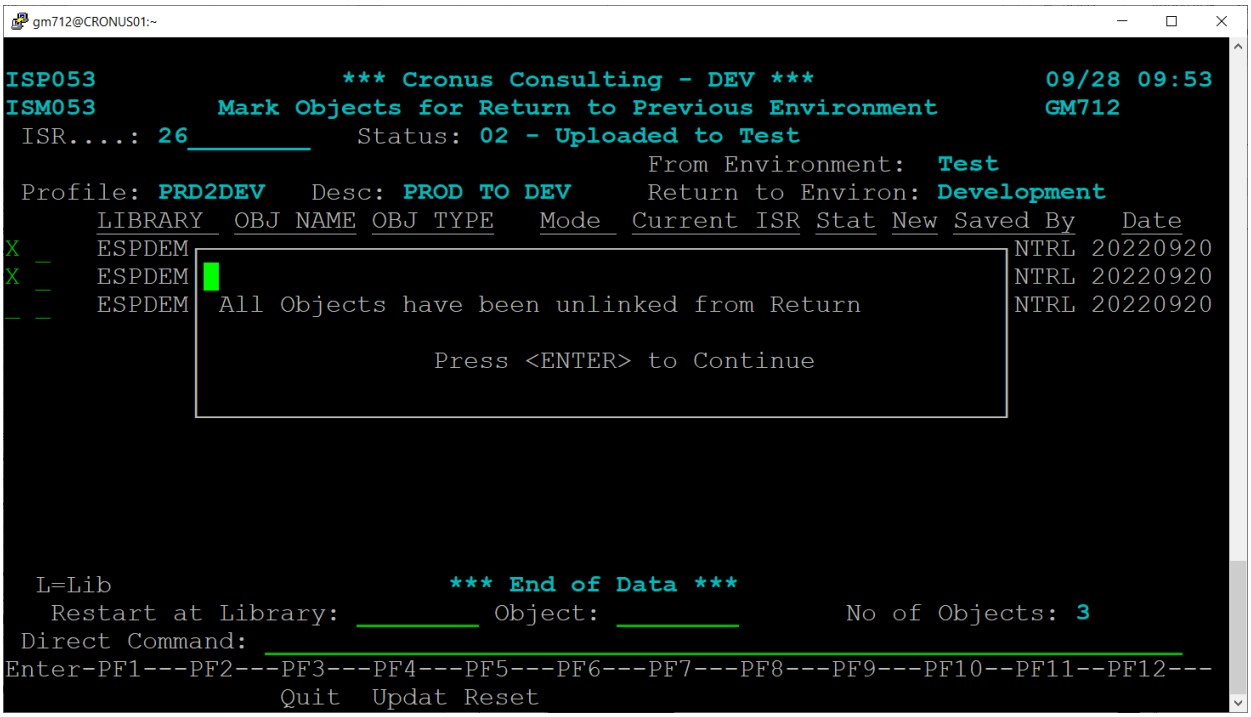


PF4 Confirmation of Return Linking Update

Objects that are returned, will be returned to the previous environment using the source or object in the current environment, in the same manner that it was moved forwards (rules of the linked profile). However, no backups of these objects are taken. The backup will be redone when moving back up the line via CC100 again, using the same version number and therefore overwriting the backup with the new source.

Multiple returns of an ISR are also allowed, without all the objects being in the same status. This means that if the objects are in Test and for example, only 2 are returned to Dev, more returns via CC053 are still allowed with other objects not already in return status, being selected and then re-running CC053 until the user requires another migration via CC100. When CC100 is run again, all the modules that are in Dev will again be moved up the line to Test until all the objects are in the same status.

Objects may also be deselected by removing the X next to the object. PF5 may be used to RESET the entire ISR from Return, i.e. un-link all linked objects, reset any Error Returns in this ISR and reset the RETURN APPROVAL (see CC061) so that normal migration may continue. After the successful resetting of the ISR, a confirmation window will be displayed as follows:

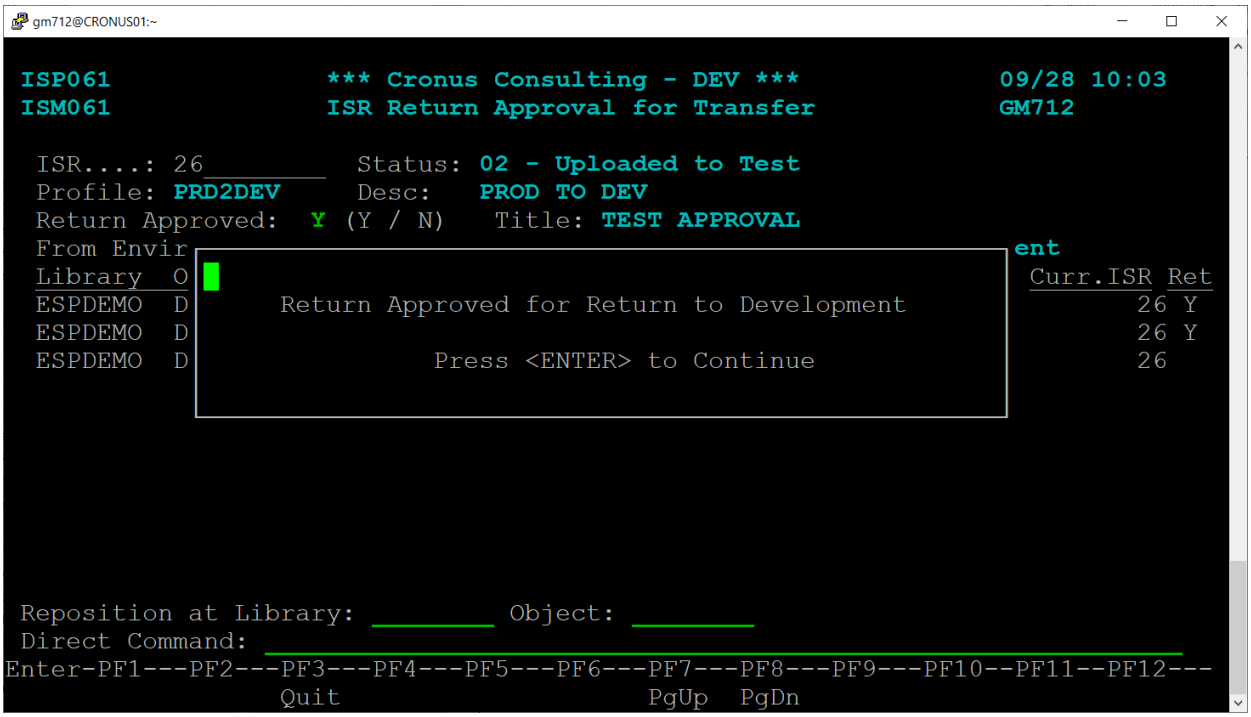


Reset Returned Objects

By marking the LIB column with an ‘L’, a window will be reflected displaying all the libraries linked to the selected object. This window is read-only, as the Return ISR transaction involves ISR’s already in use, and therefore these libraries cannot be amended here. The same Library rules will be applied as were set up in the normal migration.

3.5.2 CC061 – Return Approval for ISR

This function is used to Return Approve an ISR and to internally set the status of the ISR so that return is possible. Once an ISR has been approved for return to a previous environment, the return migration function (CC102) will be used to return all objects that have been linked for return (in CC053) to the previous environment. **NOTE: if the code RETAPP in CC010 has been marked with an "N", then CC061 can be bypassed.**

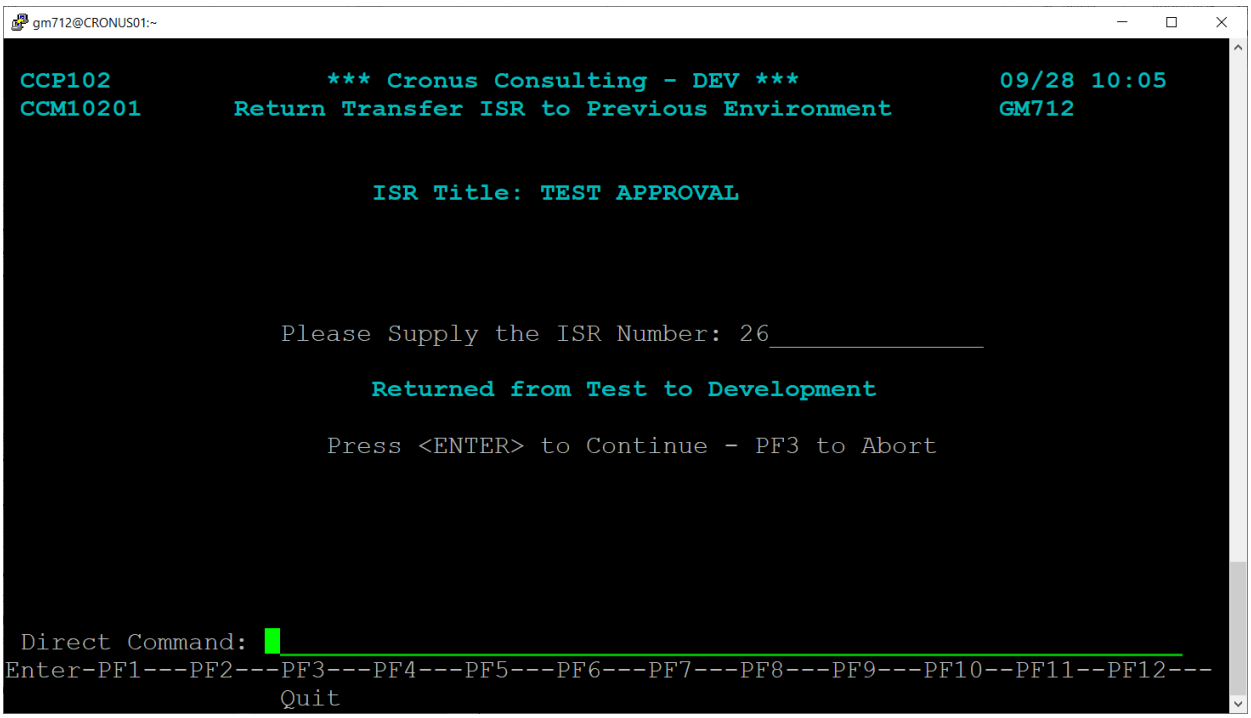


Return Approval of an ISR

3.5.3 CC102 - Return ISR to Previous Environment

This function is used to **return** all **objects marked for Return (via CC053)** linked to an ISR between the defined environments. This function is identical to the function CC100 specified above, except that the marked objects are migrated to the previous environment and no backup or restores are done. If any error occurs during the return, NO RESTORE is done. These error objects will be displayed in the ISR History enquiry and will be reflected on an error report. The function CC102 may be run again and again to return these error objects to the previous environment, until the return is successful, therefore no restore is necessary.

According to the current status of an ISR and the profile linked to it, the system will determine to which environment, objects must be returned.



Return an ISR

Once the user has pressed enter to continue the ISR will be validated for return via the normal PATH VALIDATION routine – see explanation in CC100.

```
gm712@CRONUS01:~
CCP102              *** Cronus Consulting - DEV ***              09/28 10:05
CCM10202      Return Transfer ISR to Previous Environment        GM712
  ISR NO: 26 - TEST APPROVAL
Library  Object   Backup Copy Difference        Compile NAT Err Line Resp

            Returned from Test to Development

                 Press <Enter> to Continue

ESPDEMO  DEMO02    N     Y  4     Y                S  2
ESPDEMO  DEMO03    N     Y  4     Y                S  2
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

Return ISR Confirmation

If ALLOW-SOURCE = “N”, when returning to the Initial Environment (the first environment of the profile), the source will be migrated to this environment. If ALLOW-SOURCE = “N” in any other environment, the return of objects will be the object code only.
If the ALLOW-SOURCE = “M” or “Y”, the code will always be returned to the previous environment.
If the STOW-CAT option = N, then only the source will be returned and saved, else it will be STOWed or CATalogued as selected.

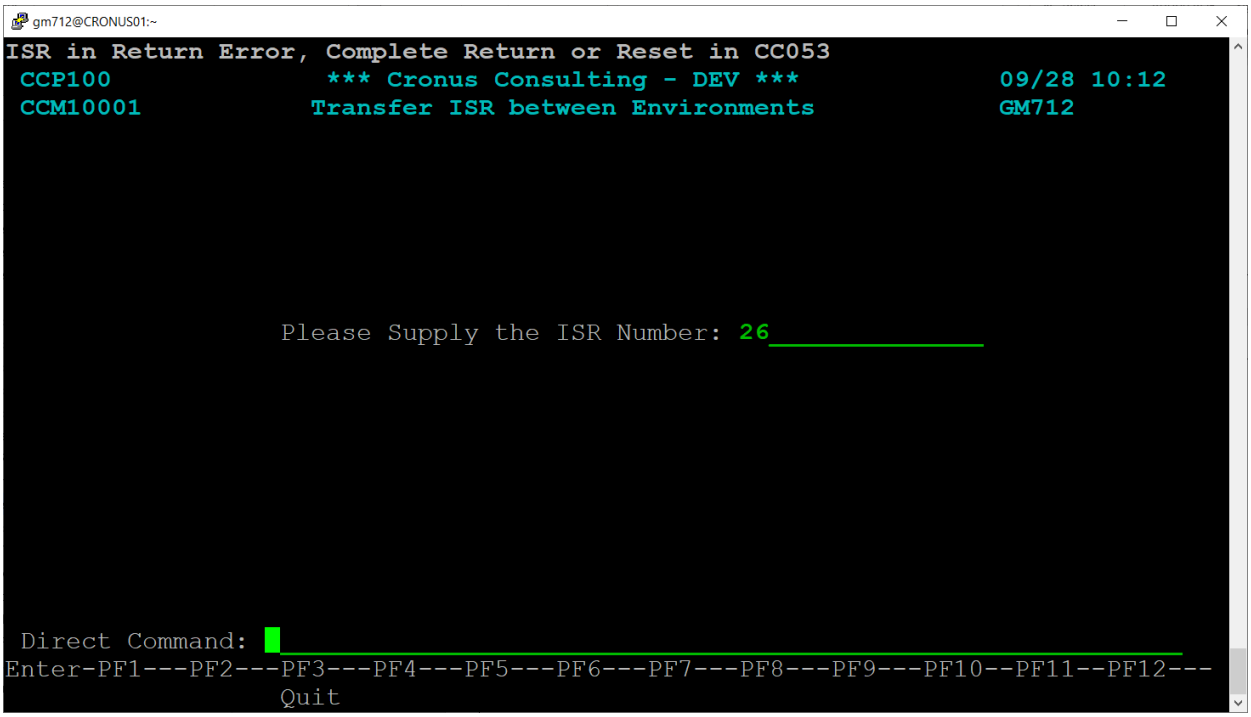
The next time the ISR is migrated it will have to move forwards again to the next environment and so on. Remember, only objects marked for Return will be returned, all unmarked objects will remain in the higher environment until all the returned objects have been migrated forward again and all objects are in the same status.
If the object is successfully returned, the object status is updated with the new environment (lower status) and is treated accordingly. If the object is unsuccessful it will remain **LINKED FOR RETURN** and the function CC102 must again be run until the object has been moved successfully. See below for Error Report example or go to the ISR History (CC095) to view the error.

```
gm712@CRONUS01:~
CCP102              *** Cronus Consulting - DEV ***              09/28 10:10
CCM10202      Return Transfer ISR to Previous Environment        GM712
  ISR NO: 26 - TEST APPROVAL
Library  Object   Backup Copy Difference        Compile NAT Err Line Resp

        UNSUCCESSFUL Return from Test to Development
                     No Restore Done
    Print Return Error Report - or go to History    Y
                Press <Enter> to Continue

ESPDEMO  DEMO04    N     Y  4     Y               E  2     0001 8888
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

Error Screen for Return

```
gm712@CRONUS01:~
Job number 311884: N.20220928.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/28                  Return Transfer ISR Error Report - ISR No -       >
10:11:27.0                   http://www.cronus.co.za

Environment From: Test
Environment To:   Development


Program    User    Date        Time                                Error Descri>
   Id       Id
-------- -------- ---------------- ----------------------------------------->

CCP102   GM712    20220928 10:10:47 Returned from Environment Test to Developme>
         GM712    20220928 10:10:47 ESPDEMO DEMO04 - Compile Error - Error No 0>



   Before continuing, fix errors and redo Return transaction
   If not required, go back to CC053 and unlink error objects

  If one of the above options is not done, ISR will be forever in Error
```

Error Report for Return



CC100 - Example of an ISR in Return Error

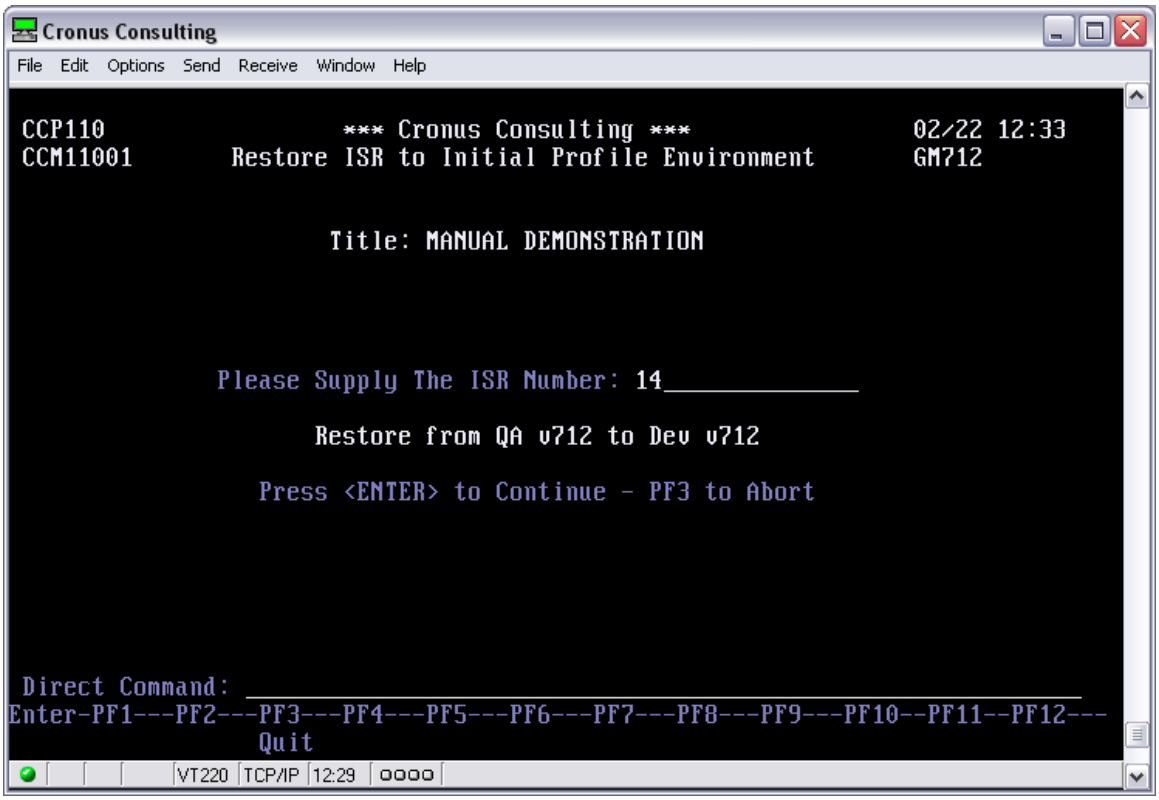
The above ISR is in return error and until a successful return has occurred via function CC102 or all the objects have been reset in CC053, the normal migration via CC100 will be disallowed.

3.5.4 CC110 - Restore ISR to Initial Environment

This function is used to restore all objects linked to an ISR, back to the initial environment of the profile linked to the ISR being restored. All other environments will be restored to the backup that was taken during the migration process of the ISR. If an environment has not yet been reached, no changes or restore will take place.

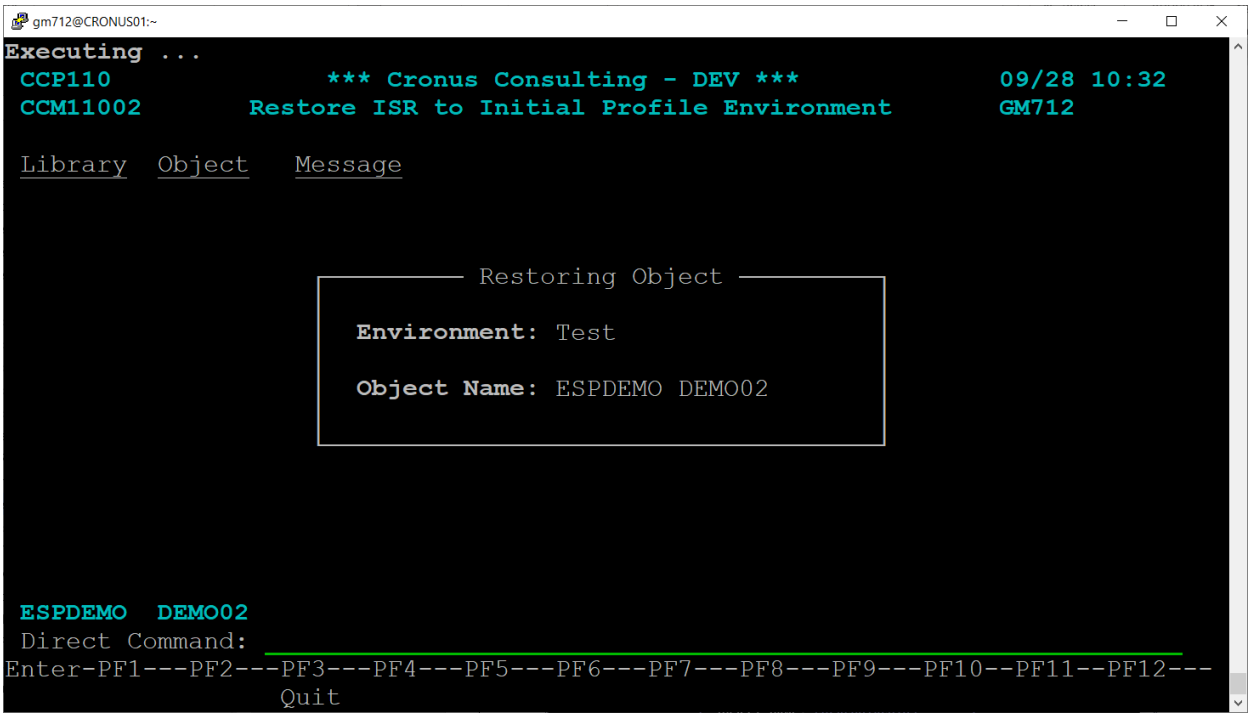
NOTE: The restore of an ISR will restore to the backups taken in the ISR (which only happen after a migration move) and so will not restore the code to the state it was in before the ISR was run. If no backup exists, then a previous backup version of the previous ISR linked to this object will be used. If this is not what is required, then function CC200 MUST be used and the correct ISR may be chosen to restore whatever source or object is necessary.

This is also library dependent on the library rules that were set up for the object when migrating via CC100. The same library per object will be used that was used in CC100. The objects will be restored in the correct object type order, so that compiling may take place correctly.



Restore Full ISR to Development

The rules specified for which backups are used in the restore, follow the backup path explained in CC100 and the restore routine as explained above, for all restore functions CC110, CC115, complete ISR CC070 with restore and unlink objects CC050 with restore.



Restoring Confirmation window

The backup version of the source code is restored in the target environment and the ISR status is updated accordingly – the object status for each object is set to the initial environment status.

If there is no backup (and no previous backup) or backup indicator is set to No, then a window reflecting “no restore of object” will be displayed. If backups exist, a window reflecting “restoring object” and what environment is being restored to, is displayed as seen in the above figure. These windows are representative of ALL restore functions and are displayed in the same manner, no matter what restore functions have been selected. See next page for example of the “no restore window”.

If backup indicators are set to “N” for the profile linked to the ISR, none of the objects in the ISR will be restored, but the object status will be set to the initial environment.

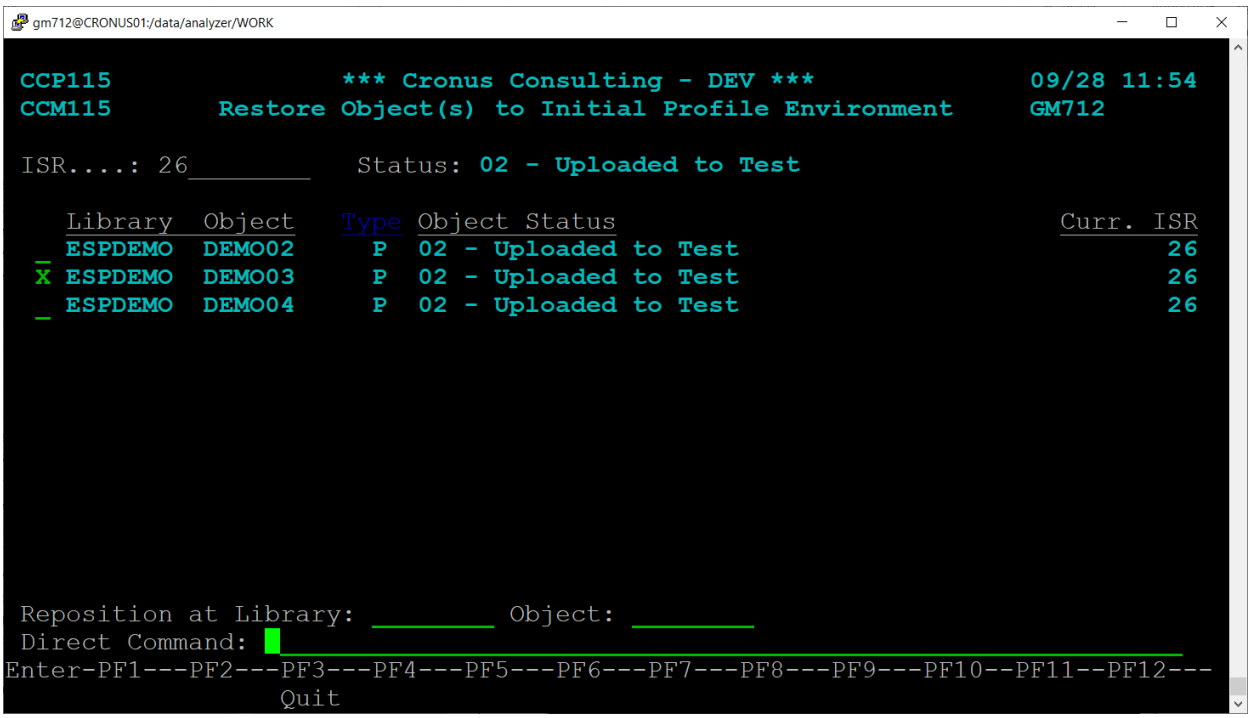
```
gm712@CRONUS01:/data/analyzer/WORK
Executing ...
 CCP110              *** Cronus Consulting - DEV ***              09/28 11:52
 CCM11002       Restore ISR to Initial Profile Environment        GM712

 Library  Object   Message


                    ──────── No Restore of Object ────────
                    │ Environment: Test                      │
                    │ Object Name: ESPDEMO DEMO02            │
                    ────────────────────────────────────────


 ESPDEMO  DEMO02
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

No Restore window

3.5.5 CC115 - Restore Object(s) to Initial Environment

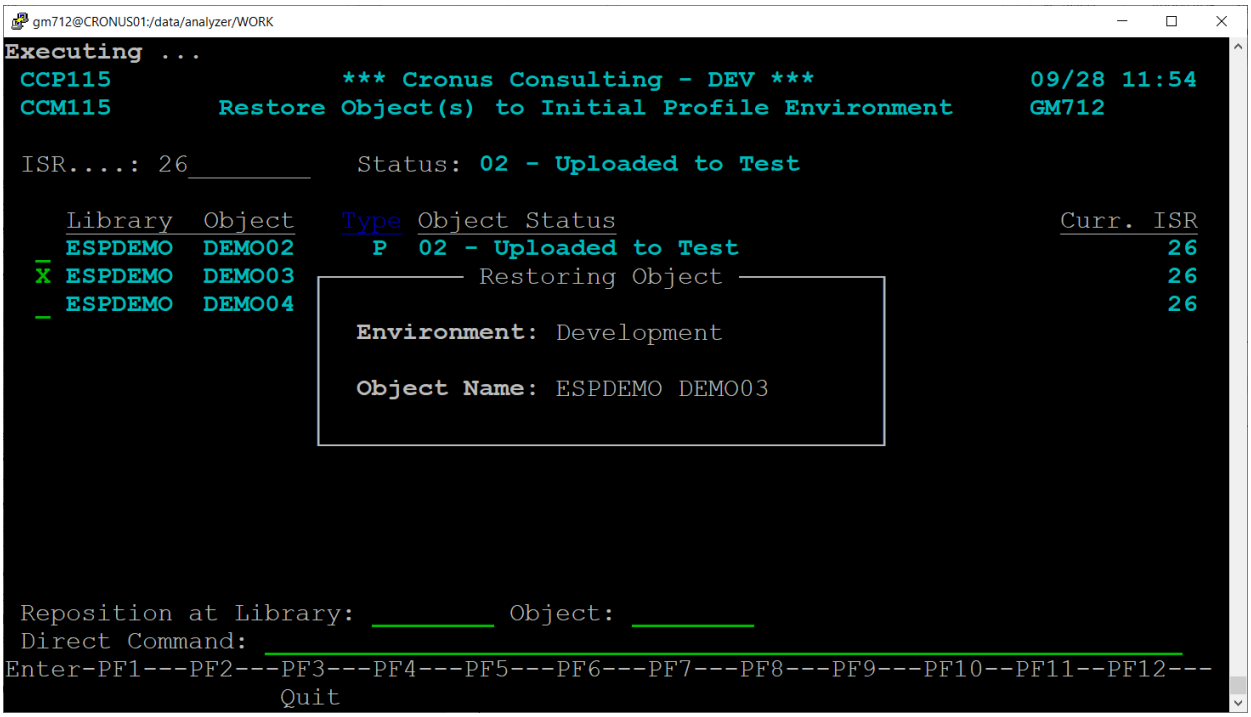
This function is used to restore individual objects linked to an ISR back to the initial environment. All environments that have been migrated to, including the initial (if in the case of no source), will be restored to the backup version, either taken in the ISR after a move, or the previous backup taken for that specific object. If the backup indicators are set to N, none of the objects will be restored, only the object status will be updated to the initial environment. **See rules for restore in function CC110, the only difference between the two functions is that CC115 allows restore of selected objects while CC100 restores entire ISR giving no choice.**

This is also library dependent using the same rules as was set up when first running CC100 for a particular ISR.



Object Selection for Restore

Confirm the restore with <enter>



Restoring Confirmation Window

See CC110 for example of NO RESTORE window, if no backups exist or backup indicator is set to N.

NOTE : As this is a manual restore function, the correct object types must be selected first e.g. maps, copy-code etc for restore, else the compile of the restored object will not take place correctly and only the source will be SAVEd. If the entire ISR is to be restored, rather use function CC110, which restores objects in correct object order. The unlinking of objects in CC050, also has the same manual restore effect if, unlinking with restore, is selected.

3.5.6 CC200 - Restore Objects Versions to ANY Environment in ISR

The main purpose of this function is used to Display or Restore previous object versions using the backed up copy via the EspControl System per ISR. The number of object versions available is dependent on the system default for Maximum backups set in CC001. If No backup is selected for specific ISR's, then obviously no restore or display may be done. This function is available to restore to or display source of any ISR (as long as backup still exists) across any of the environments that were used during the migration of that specific ISR. If a specific restore of objects is required, then this is the function to use, as the user may manually decide what to restore to.

The other purpose of CC200 depending on the codes ADHOCL, SCLADHOC, RESTOREPRD and EMERGENC is the C (Compare), A / PF5 /PF6 (Adhoc) and E (Emergency) options where these options all transfer source between the Development and Production environments without the creation of an ISR. There is security around these options as well. These options will be detailed below after the Display and Restore explanations.

For Display or Restore select the object with 'X' and then once selected use 'D' or 'R' against the ISR that is being selected. Once the ISR is selected, use 'X' to select the environments required. Enter to continue.

The display function across all environments enables one to first check the code before doing a restore. The display function will display the code on the screen, with the version number, environment name and object name at the top of the window. Use PF3 to exit. Enter ? on the command line and a list of commands will be displayed. Use these to page backwards and forwards and even scan for a specified piece of code. The scan function is not case sensitive and so will find the scan-set in both upper and lower case.

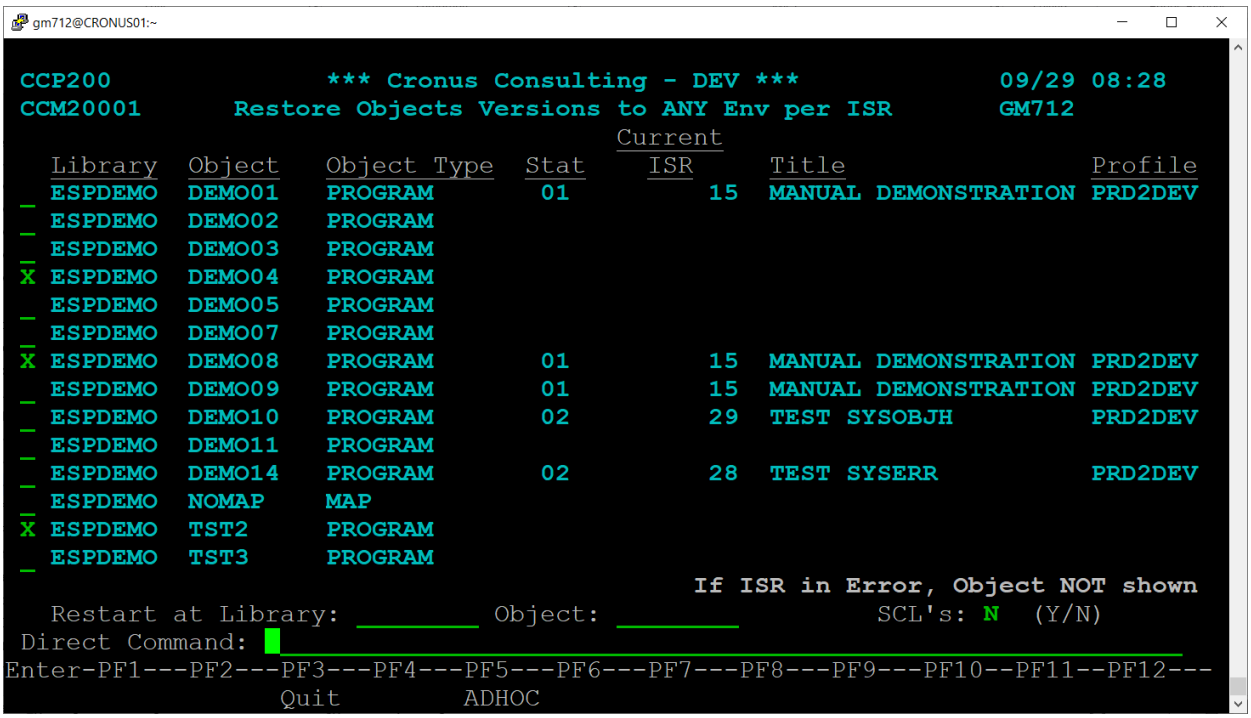
The default is to restore to the backed-up library, or the user may enter a specific library where the code must be restored to. This library is validated per selected environment and an error will be returned if the library is not valid. For example, library RESTORE can be used if you do not want the code to be directly restored to the backup library and can then use SYSMAIN to copy to the correct library.

A list of all objects in the Inventory List is displayed for the user to select, starting with the initial scan library set up in CC001 (more than one object may be selected at any given time). Once selected, the linked ISR's will be displayed. If the selection field is not open for any specific ISR, it means that the backups no longer exists for those ISR's, due to no backup taken or backups were deleted due to maximum number of versions set up in CC001, and therefore no DISPLAY or RESTORE can be done. The number of maximum versions is therefore of great importance for the site to decide on when and how to restore.

Once an ISR No has been selected, the environments for that ISR (as were set up from the linked profile) will be reflected and the user may select one or all of the environments, depending on the type of restore

or display. Once restore has been confirmed and relevant restore library selected by the user, the restore will commence. If only Display has been selected, code will be displayed on the screen. Once complete for that particular object use PF3 to automatically continue to the next selected object or the same object in the next selected environment.

The backup path is determined from the backup-history file and CC200 will check for backups in this directory.



Object Selection for Display/Restore

Select the object(s) to see available versions saved for this object.

The following information is given per object:

- Library - Library where object resides (Scanned in Library)
- Object Name - Object name
- Object Mode - Object programming mode (S/R)
- Object Type - Object type description
- Object Status - Object status - Blank if not linked
- Current ISR No - Current ISR nr - Blank if not linked to an ISR
- ISR Title - ISR Title - Blank if not linked to an ISR
- Profile - Profile Name if linked to an ISR

Note: The restore compiles the object if applicable. In other words, it will try and compile the source if it has been restored to a library where the object has all the resources to stow.

```
gm712@CRONUS01:~

CCP200              *** Cronus Consulting - DEV ***              09/29 09:09
CCM20002      Restore Objects Versions to ANY Env per ISR        GM712

Library: ESPDEMO  Object: TST2       Obj Env: Development

     ISR No ISR Title            Originator          Completed  Developer
_        14 INITIAL BACKUP       GM712               2022-09-14 GM712
D        13 INITIAL BACKUP       GM712               2022-09-14 GM712
_         9 TEST                 GM712               2022-09-09 GM712
_         8 TEST                 GM712               2022-09-08 GM712

Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        ADHOC
```

ISR Selection for Restore or Display (D or R)

Function Options:

D - Display Object

R - Restore Object

The following information is given per object:

- Current ISR No - Current ISR nr - Blank if not linked to an ISR
- ISR Title - ISR Title - Blank if not linked to an ISR
- Originator - Originator of ISR
- Completed - Date selected ISR was completed
- Developer - Developer/Person that modified the code

A window will next be displayed of the environments of the selected ISR. It is for this reason that environments may **not** be deleted from CC001, as they may be needed for further use in a restore. If an environment is no required, then mark PATH = N, for all current profiles still in use. This will not allow any further migrations to these environments.

```
gm712@CRONUS01:~
CCP200              *** Cronus Consulting - DEV ***              09/29 09:09
CCM20002      Restore Objects Versions to ANY Env per ISR        GM712

Library: ESPDEMO  Object: TST2       Obj Env: Development

     ISR No IS                                                   per
         14 IN Select Environment(s) with X
D        13 IN  Sel Environment
          9 TE   X  Development
          8 TE      Test
                 X  Production

               Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---P
                                 Exit                     PgUp  PgDn

Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit        ADHOC
```

Select Environment for Restore or Display

All available environments linked to the selected ISR, as determined by the profile for that ISR, will be displayed and the user may select with “X” more than one or a subset for restore or display.

```
gm712@CRONUS01:~
LIST: TST2 VER: 3 - Development  - PF3 TO EXIT
0010 * This is a test pgm for ESPDEMO
0020 END
0030 * 20190628 ISR: 0000023 Desc: TEST GP MOVES        Changed By: GM712
0040 * 20220908 ISR: 0000008 Desc: TEST                 Changed By: GM712
0050 * 20220909 ISR: 0000009 Desc: TEST                 Changed By: GM712

ISR       : 13
File Path: /opt/softwareag/Natural/fuser_natdbck/.BCK/ESPDEMO/SRC/TST2.NSP.0
Command   :                                                    (? - Help)
```

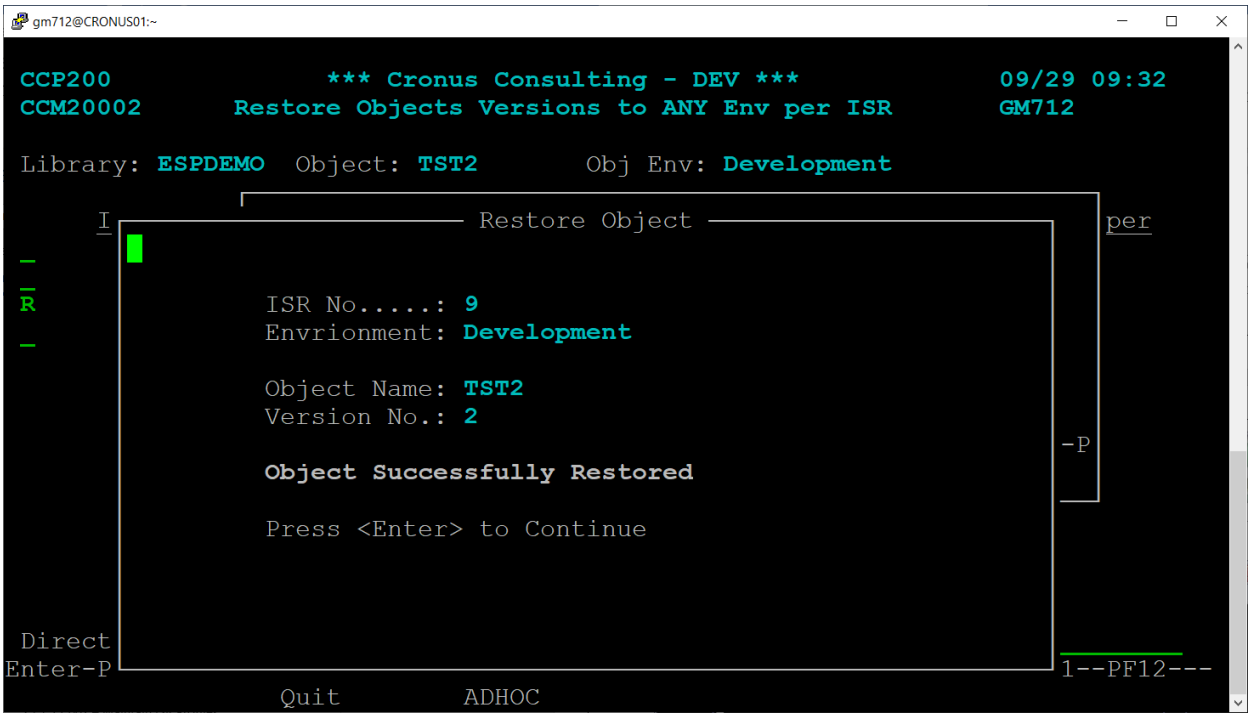
Display Source Window

```
gm712@CRONUS01:~
LIST: TST2 VER: 3 - Development  - PF3 TO EXIT
0010
0020
0030   ?, HELP                    - DISPLAY HELP                    12
0040   --, T, TOP                 - TOP                             12
0050   ++, B, BOT, BOTTOM         - BOTTOM                          12
       -                          - PAGE BACK
       +                          - PAGE FORWARD
       - H                        - HALF A PAGE BACK
       + H                        - HALF A PAGE FORWARD
       NNNN                       - RESTART AT LINE NNNN
       SC <SCAN VALUE>            - SCAN FOR <SCAN VALUE>
       1 SC <SCAN VALUE>          - SCAN FOR <SCAN VALUE> FROM TOP
       PF3                        - EXIT
       PF5                        - REPEAT SCAN
       PF7, PGUP                  - PAGE BACK
       PF8                        - PAGE FORWARD
       PF9                        - REPOSITION AT CURSOR
       ENTER                      - SCROLL FORWARD 1 LINE
 ISR
 File                                                          .NSP.0
 Command  : ?                                            (? - Help)
```

List of commands for User to enter on Command Line

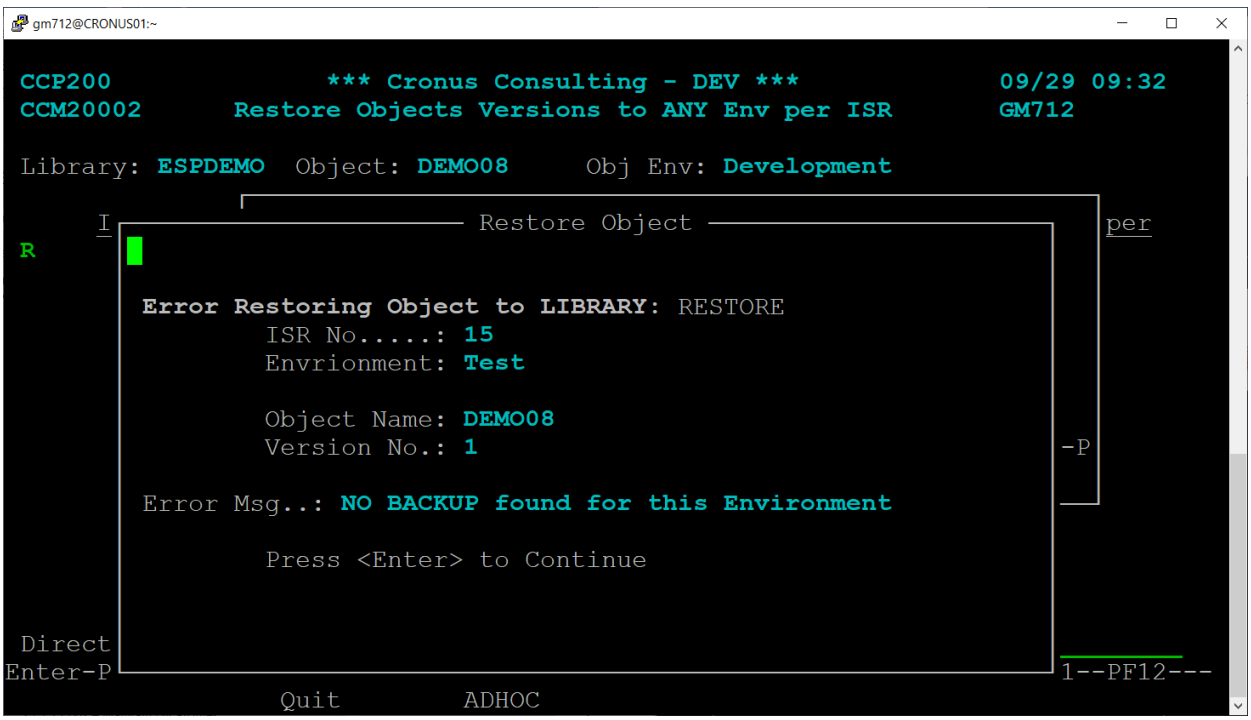
```
gm712@CRONUS01:~
CCP200          *** Cronus Consulting - DEV ***          09/29 09:09
CCM20002     Restore Objects Versions to ANY Env per ISR   GM712

Library: ESPDEMO  Object: TST2      Obj Env: Development
     I                                                        per
                    Restore Object
_
_      Source will be restored to Backup LIBRARY: ESPDEMO
R      Source will be STOWED, if applicable
_
               ISR No......: 9

               From Library: ESPDEMO
               Object Name.: TST2                            -P

               Environment.: Development

               Confirm.....: Y (Y)es or (N)o

       Amend Library ID to restore to another ESPDEMO_
 Direct
Enter-P                                                  1--PF12---
                Quit        ADHOC
```

Restore Details per ISR



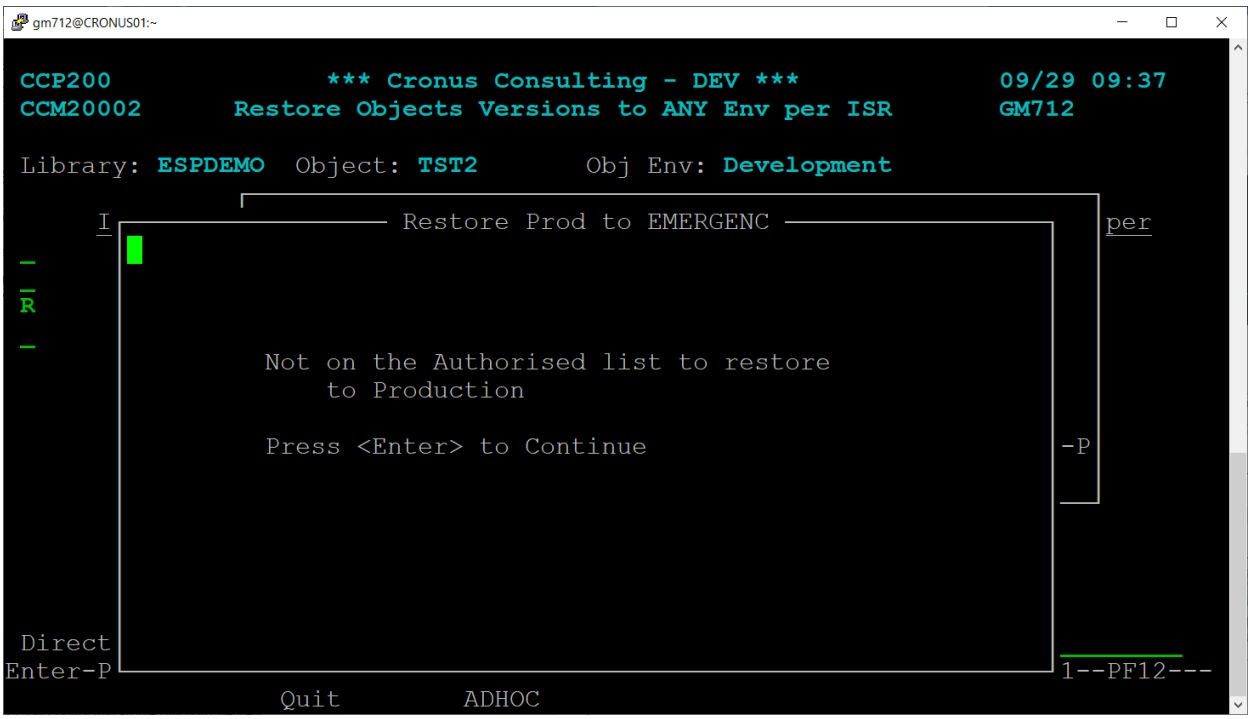
Restore Confirmation per ISR

See example of error window below if no backup exists, so no source can be restored. The same would be applicable if Display was selected.



No source backup so No Restore

Code RESTOREPRD can be used to limit the users who can restore to Production. If the code RESTOREPRD only has the default PRDUSER code value, then all who have access to CC200 can restore. See example below where the RESTOREPRD has users defined, and therefore the Restore function is limited.



RESTOREPRD code in use, and authorised list checked for Production

Code RULES and sub-code EMERGENCY is used to determine if the object being restored to Production should go to the EMERGENCY library. This will default to the EMERGENCY library if set to Y.

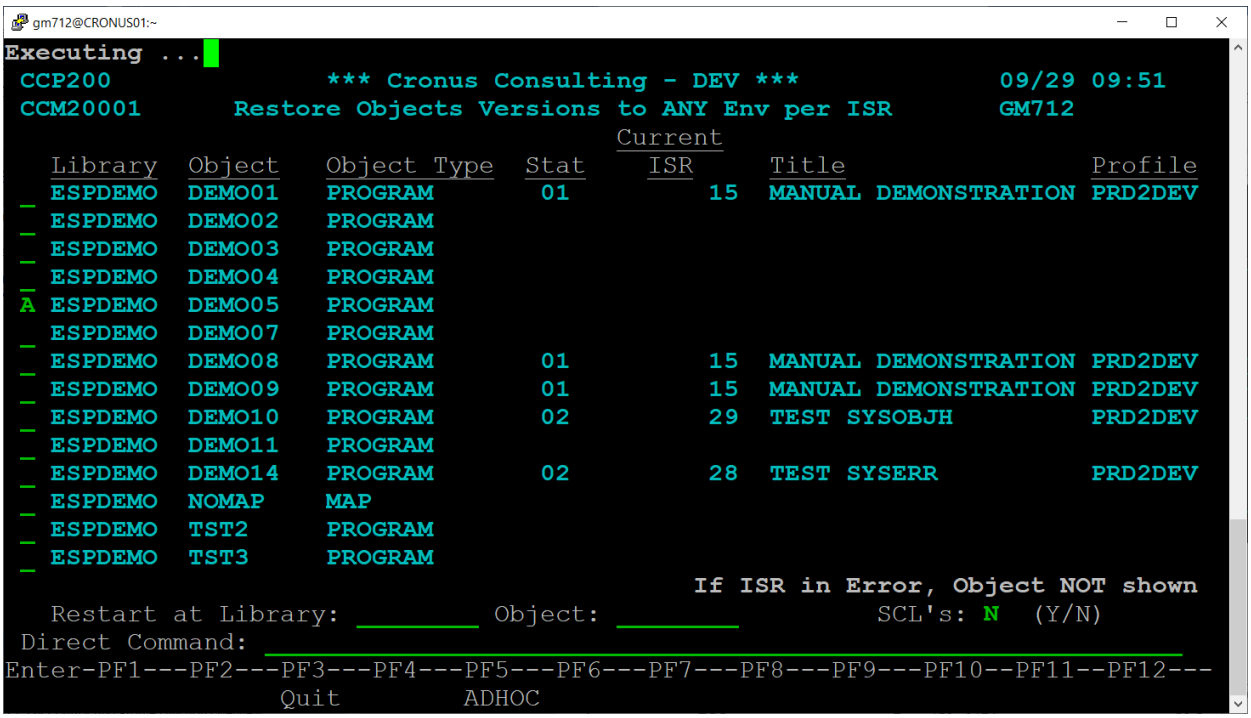
Option ADHOC - Code ADHOCL is used to set the library that will be used for ADHOC object copies. Option A next to an object or PF5 is used for ADHOC copies. An ADHOC copy will directly migrate the source from Development to Production without the need of an ISR. The ADHOC library set in the 1st code will be used as the library in the ADHOC function. The ADHOC library must be set up in Natural Security in Development if Natural Security is used, else the copy will not work. Option A can be used next to the object, or of the object is not on the normal Inventory List it can still be moved using PF5.

The ADHOC function will only be allowed if the 2nd code = 'E' or "Y'. If it is set to N, then option A and PF5 will not be allowed. The 3rd code is the Production environment number and the 4th code is the Development environment number. (See detailed explanation in CC010).

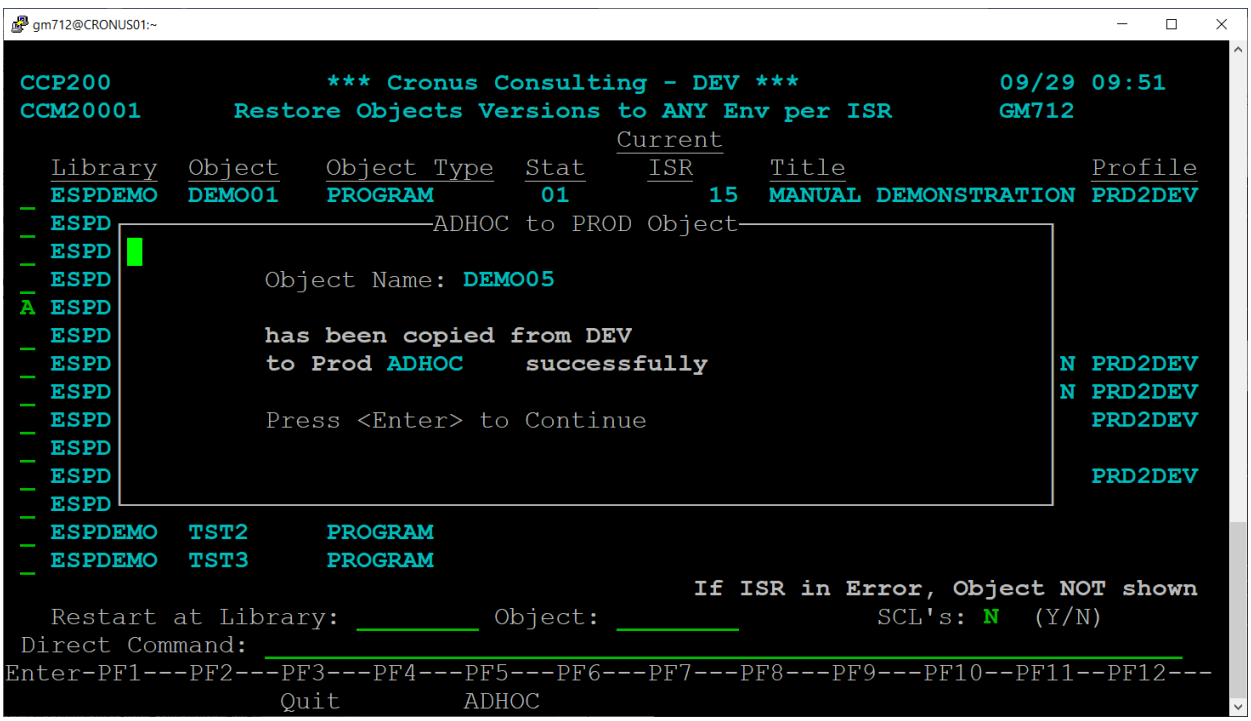
If the 2nd code is set to 'E', then if code RESTOREPRD has a set of users, the ADHOC function will be limited to this set of users.

The ADHOC function will use the environment numbers in code ADHOCL to determine the development and production environment.

CC095 keeps a simple log of all ADHOC copies (both object and SCL) and can be seen when selecting PF6 which is always linked to ISR 77777777.

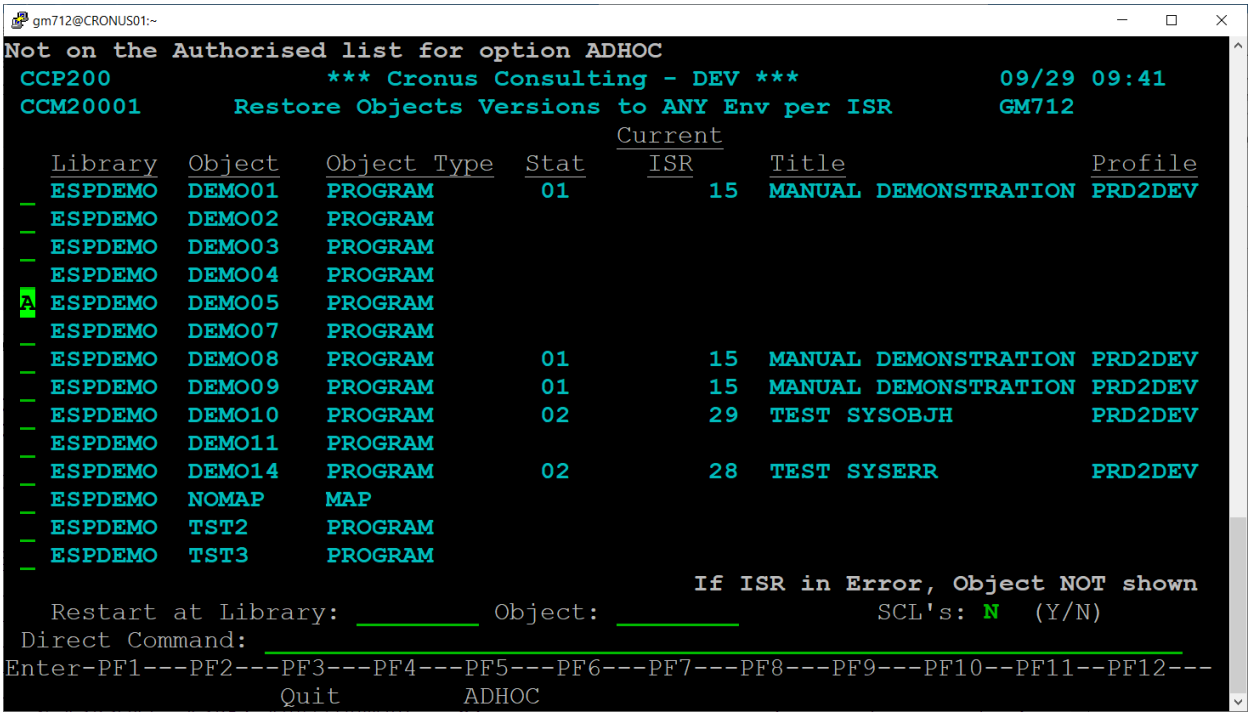


Option A for ADHOC function

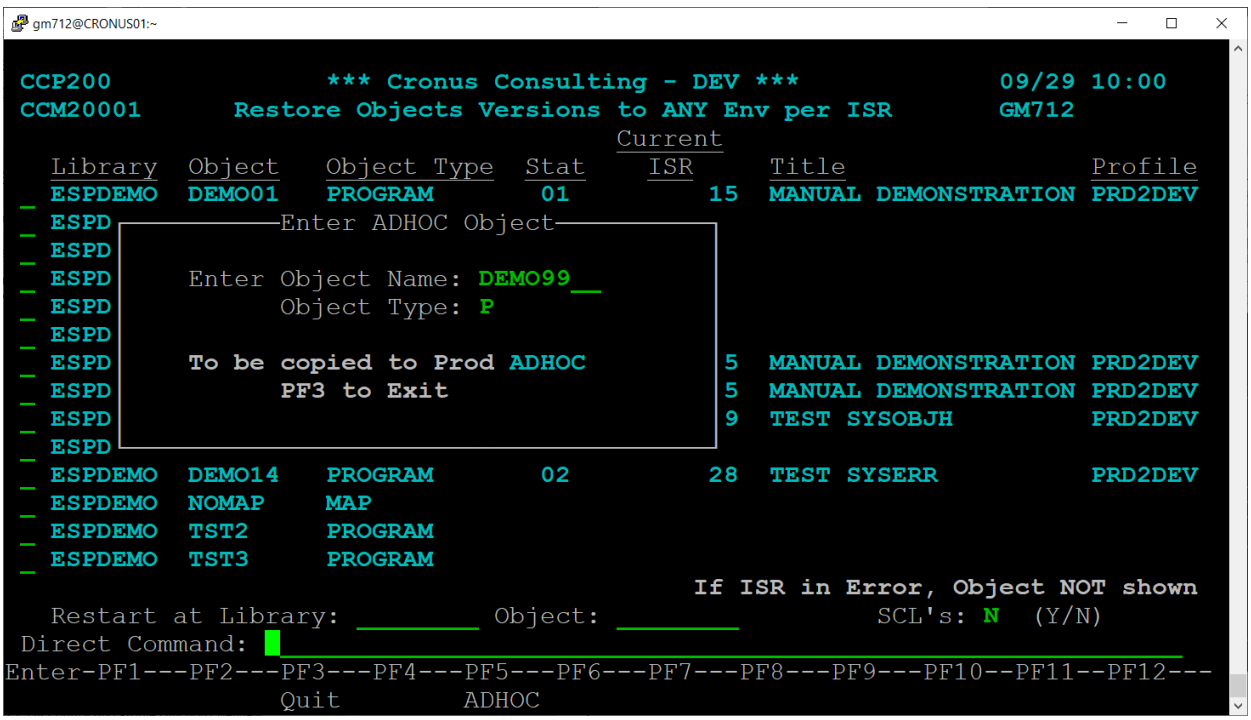


DEMO05 has been copied to library ADHOC

The library ADHOC as seen above can be amended on the code ADHOC in CC010. The copied source is compiled. If the object cannot be compiled successfully, it will just copy the source.

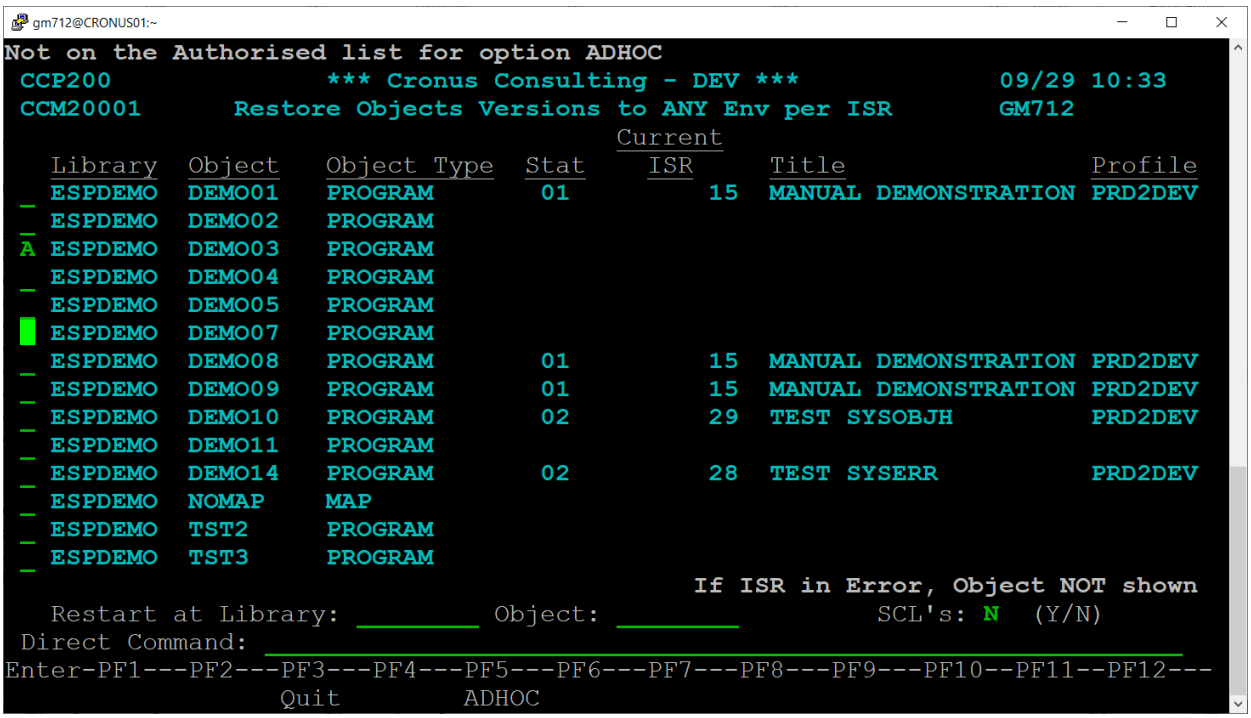


RESTOREPRD code in use, and authorised list checked for ADHOC option (A or PF5)



Using PF5 for ADHOC

Using PF5 for ADHOC requires the object name and type to be input. This is an option when you do not want to scan the module into the Inventory List. A check will be done to ensure that the object does exist in the ADHOC library.



ADHOC option restricted to list of users in RESTOREPRD

Option COMPARE - Option C next to an object is used for COMPARE copies. A COMPARE copy will directly migrate the source from Production to Development to library COMPARE without the need of an ISR. Option C is available to all users of CC200 and is used to compare Development to Production source especially if using NaturalOne. NaturalONE has a compare function that can be used between two libraries. Library COMPARE must be set up in Natural Security in Development if Natural Security is used, else the copy will not work. The COMPARE copy will copy the module from Production using the library in CC200 to a static library called COMPARE. This library cannot be changed.

CC095 keeps a simple log of all COMPARE copies and can be seen when selecting PF2 which is always linked to ISR 99999999.

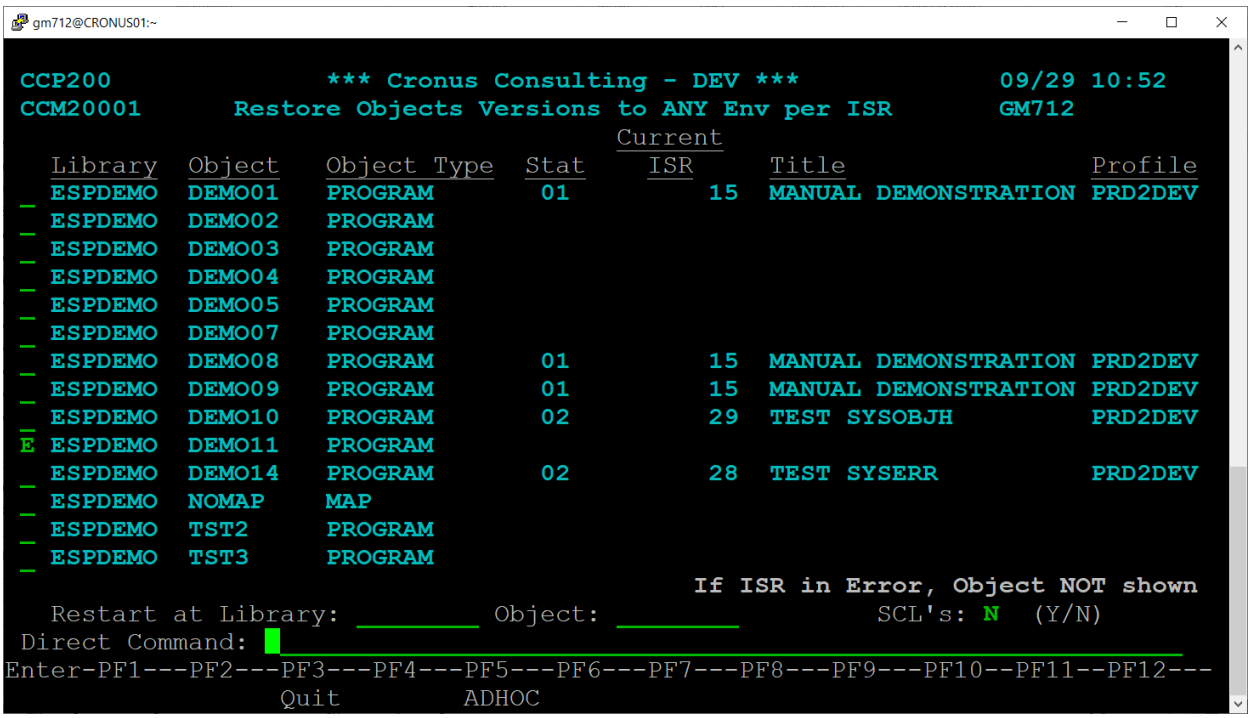
```
gm712@CRONUS01:~
Executing ...
 CCP200              *** Cronus Consulting - DEV ***              09/29 10:44
 CCM20001      Restore Objects Versions to ANY Env per ISR        GM712
                                          Current
   Library   Object    Object Type   Stat    ISR      Title                Profile
   ESPDEMO   DEMO01    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
   ESPDEMO   DEMO02    PROGRAM
   ESPDEMO   DEMO03    PROGRAM
   ESPDEMO   DEMO04    PROGRAM
   ESPDEMO   DEMO05    PROGRAM
 C ESPDEMO   DEMO07    PROGRAM
 _ ESPDEMO   DEMO08    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
 _ ESPDEMO   DEMO09    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
 _ ESPDEMO   DEMO10    PROGRAM        02        29  TEST SYSOBJH         PRD2DEV
 _ ESPDEMO   DEMO11    PROGRAM
 _ ESPDEMO   DEMO14    PROGRAM        02        28  TEST SYSERR          PRD2DEV
 _ ESPDEMO   NOMAP     MAP
 _ ESPDEMO   TST2      PROGRAM
 _ ESPDEMO   TST3      PROGRAM
                                             If ISR in Error, Object NOT shown
   Restart at Library: ________ Object: ________          SCL's: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        ADHOC
```

Option C for COMPARE function

```
gm712@CRONUS01:~
 CCP200              *** Cronus Consulting - DEV ***              09/29 10:44
 CCM20001      Restore Objects Versions to ANY Env per ISR        GM712
                                          Current
   Library   Object    Object Type   Stat    ISR      Title                Profile
   ESPDEMO   DEMO01    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
   ESPD ┌──────────────────────── COMPARE LIB Object ──────────────────────┐
   ESPD │
   ESPD │          Object Name: DEMO07
   ESPD │
 C ESPD │          has been copied from PROD
 _ ESPD │          to DEV COMPARE successfully                            N PRD2DEV
 _ ESPD │                                                                 N PRD2DEV
 _ ESPD │          Press <Enter> to Continue                                PRD2DEV
 _ ESPD │
 _ ESPD │                                                                   PRD2DEV
 _ ESPD └──────────────────────────────────────────────────────────────────
 _ ESPDEMO   TST2      PROGRAM
 _ ESPDEMO   TST3      PROGRAM
                                             If ISR in Error, Object NOT shown
   Restart at Library: ________ Object: ________          SCL's: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        ADHOC
```

DEMO07 has been copied to library COMPARE

Option EMERGENC - Option E next to an object is used for EMERGENCY copies. An EMERGENCY copy will directly migrate the source from Production to Development to the library set in the EMERGENCY code in CC010, without the need of an ISR. Option E is available to all users of CC200 and is used to copy objects from the library in CC200 to the EMERGENCY library. This is defaulted to EMERGENC in code EMERGENCY in CC010, but can be changed to any emergency type library. The EMERGENCY library must be set up in Natural Security in Development if Natural Security is used, else the copy will not work. CC095 keeps a simple log of all EMERGENCY copies and can be seen when selecting PF2 which is always linked to ISR 99999999.



Option E for EMERGENC function

```
gm712@CRONUS01:~
CCP200              *** Cronus Consulting - DEV ***              09/29 10:52
CCM20001      Restore Objects Versions to ANY Env per ISR        GM712
                                        Current
  Library   Object    Object Type   Stat    ISR     Title                Profile
_ ESPDEMO   DEMO01    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
_ ESPD ┌──────────────────────────EMERGENC LIB Object──────────────────────┐
_ ESPD │
_ ESPD │          Object Name: DEMO11
_ ESPD │
_ ESPD │          has been copied from PROD
_ ESPD │          to DEV EMERGENC successfully                            N PRD2DEV
_ ESPD │                                                                  N PRD2DEV
_ ESPD │          Press <Enter> to Continue                                 PRD2DEV
E ESPD │
_ ESPD │                                                                    PRD2DEV
_ ESPD └──────────────────────────────────────────────────────────────────┘
_ ESPDEMO   TST2      PROGRAM
_ ESPDEMO   TST3      PROGRAM
                                                  If ISR in Error, Object NOT shown
  Restart at Library: ________ Object: ________          SCL's: N   (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit        ADHOC
```

DEMO11 has been copied to library EMERGENC

```
gm712@CRONUS01:~
Valid Options are: (C)ompare, (A)dhoc, (E)mergency or 'X' to Select
 CCP200              *** Cronus Consulting - DEV ***              09/29 11:01
 CCM20001      Restore Objects Versions to ANY Env per ISR        GM712
                                        Current
   Library   Object    Object Type   Stat    ISR     Title                Profile
 _ ESPDEMO   DEMO01    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
 _ ESPDEMO   DEMO02    PROGRAM
 _ ESPDEMO   DEMO03    PROGRAM
 _ ESPDEMO   DEMO04    PROGRAM
 _ ESPDEMO   DEMO05    PROGRAM
 _ ESPDEMO   DEMO07    PROGRAM
 _ ESPDEMO   DEMO08    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
 _ ESPDEMO   DEMO09    PROGRAM        01        15  MANUAL DEMONSTRATION PRD2DEV
 _ ESPDEMO   DEMO10    PROGRAM        02        29  TEST SYSOBJH         PRD2DEV
 _ ESPDEMO   DEMO11    PROGRAM
 _ ESPDEMO   DEMO14    PROGRAM        02        28  TEST SYSERR          PRD2DEV
 _ ESPDEMO   NOMAP     MAP
 _ ESPDEMO   TST2      PROGRAM
 _ ESPDEMO   TST3      PROGRAM
                                                  If ISR in Error, Object NOT shown
   Restart at Library: ________ Object: ________          SCL's: N   (Y/N)
  Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        ADHOC
```

CC200 valid options if codes applicable set to Y

Appendix A – SCL Transfer

SCL transfer will only be available if EspBatch is set to 'Y' at installation of EspControl. Obviously, EspBatch cannot be set to 'Y' if no EspBatch is actually running. This will in turn create a code ESPBATCH in CC010 with a code value of 'Y' and an extra code value of 'SCL' for code ISRTYPE in CC010. Both of these codes need to be set up if SCL transfer is required.

Once set up, if SCL Transfer is no longer required, delete code value 'SCL' on code ISRTYPE and amend code ESPBATCH to 'N'.

All EspControl routines that allow SCL access, will be discussed below. If no SCL access is applicable, these screens will not be affected at all, no SCL detail or functions will be displayed, and will execute as specified in the EspControl manual above.

The functions specified below, will detail the SCL function on the screen, and unless the process is different to the function process explained above in the EspControl manual, the execution and rules for the function will remain as explained for objects, and will not appear in the Appendix A at all.

An SCL ISR and a normal ISR may not be mixed together i.e. an SCL ISR will have a different ISR number to a normal object ISR. SCL ISR's must always be linked to ISR Type 'SCL' and this will be the indicator the EspControl system needs to determine whether it is an SCL or normal ISR function. Due to the fact that specific rules are required for SCL profiles (for e.g. Move source or No source cannot be selected), these ISR's must also be linked to their own profiles. Every SCL profile must start with SCLxxx and this must always be linked to SCL type ISR's.

The backup cycle is completely different for an SCL ISR compared to a normal ISR. See explanation in manual above for backup cycle for normal ISR's. The backup of a SCL is always done BEFORE the move to the next environment, and not after the move as is done in normal ISR's. For this reason, no initial backup is required for SCL's. i.e. no CC350 must be executed for SCL's. No previous backup may be requested when unlinking a SCL from CC050, as the current backup will always be for the current ISR. Therefore, when selecting any restore function, the SCL will be restored to a version of the SCL before it was overwritten in the last migration.

Each point specified below, will use the same number conventions as used in the above manual, for ease of use and will only detail any differences. All other explanation and details may be taken from the EspControl manual as detailed above for Normal ISR's.

These ISR's may be linked and run simultaneously, or they may be executed individually at the user's discretion. The SCL's must be scanned in separately to the normal objects. This will be done for new SCL's via CC050 as with objects, but for the initial scan it must be done via a separate function – CC330. This is the only function that has been separated from normal ISR's. Every other function will detect whether it is an SCL ISR or a normal ISR and execute accordingly.

```
gm712@CRONUS01:~

 ISP090              *** Cronus Consulting - DEV ***            09/29 11:03
 ISM09002            Display Objects Linked to an ISR           GM712

 ISR....: 31_________   Status: 99 - Completed
 Profile: SCL2DEV       SCL PROD 2 DEV
'X' SCL User    Name    SCL Type Object Status                        Curr. ISR
 _  DEMO      DAILY     SCL      99 - Completed
 X  DEMO      J505S011  SCL      99 - Completed
 _  DEMO      JOB3      SCL      99 - Completed

                        *** End of Data ***
 Reposition at SCL User: ________ SCL Name: ________
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit                          PgDn
```

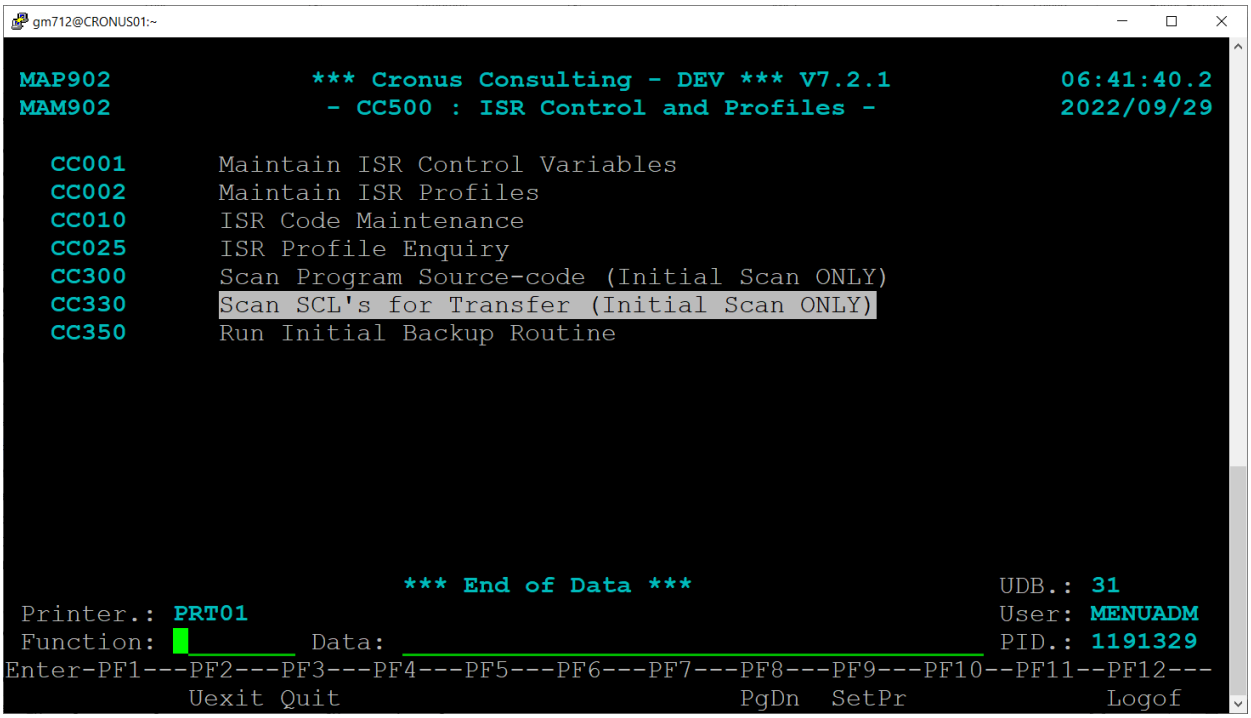
Example of how SCL's will be reflected – per user, per name and per type

Note, if an environment includes entries in CC001 for the same fuser, this must not be included in a profile for an SCL. i.e. only one entry must be marked as a PATH for an SCL. This is because, when migrating normal objects, they may move from one Natural Library to the next, in the same fuser, thereby allowing "duplicate fuser's" with different environment names. Because an SCL is not linked to a Natural environment, it will only occur once in each fuser in EspBatch under JS300.i.e there is one ESPSOFT Natural library per environment. Marking duplicate fuser's will effectively do nothing as the same SCL will overwrite itself.

If the function is to work as is, no further detail will be given, and will not be specified below at all. Note, not all screens will be shown. If an SCL ISR is allowed, the screen will function in the same manner as shown in the screen image below. For objects – Object Name and Library will be reflected. For SCL's – SCL User, SCL Name and SCL Type will be reflected, but the process of the function will remain the same.

1. Menu Overview And Function Overview showing SCL's

Sub Menu CC500



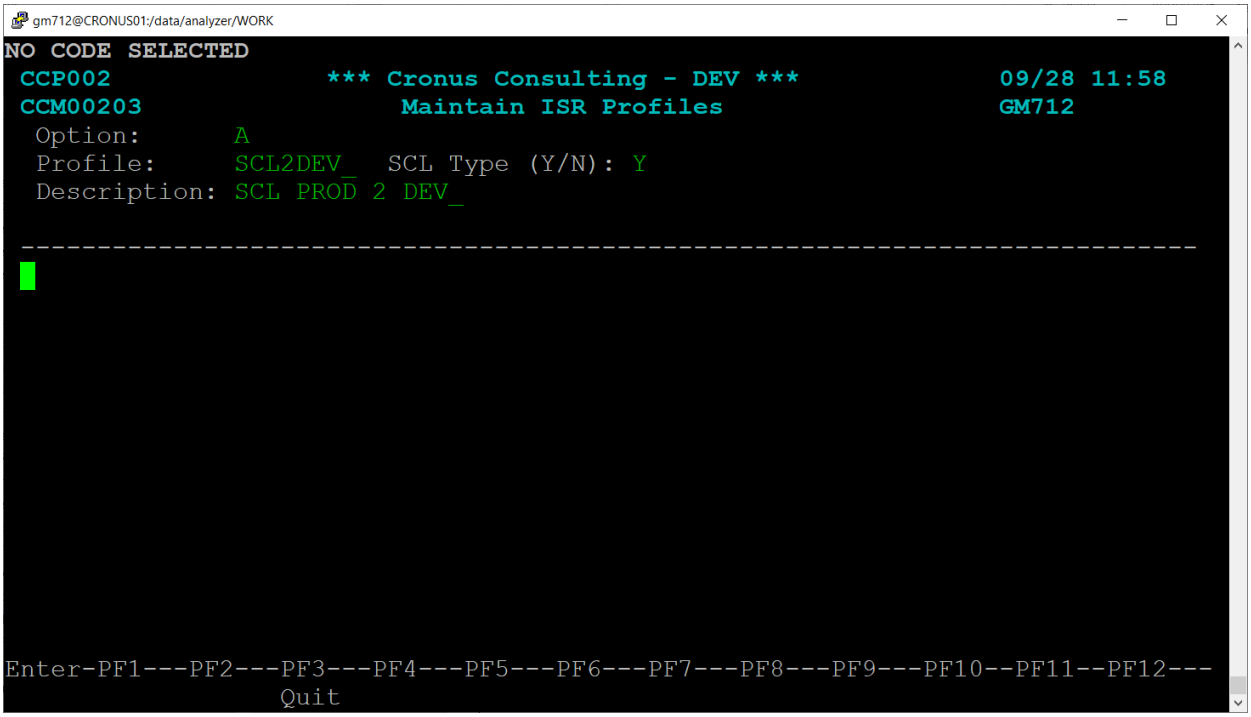
CC500 showing specific SCL function CC330

CC330 – function for scanning in SCL's. This will only be visible on sub menu CC500, if SCL transfer is allowed. The scanning of new SCL's must be done from the Production environment to set up the correct Inventory List visible in CC050. Any new SCL will be scanned via PF6 in CC050 in the Development environment.

3.1 ISR Control And Profile Function Overview

3.1.1 CC002 – Maintain ISR Profiles

When creating a new profile, the user will be requested to indicate if this is an SCL profile or a normal profile. Once created, this profile type will be displayed on the CC002 maintenance screen, but cannot be amended. If an error has been made about the type of profile, delete the profile and recapture.



CC002 – SCL Type Y or N

- For normal type profiles, mark SCL Type as N
- For SCL type profiles, mark SCL Type as Y
- If SCL type marked as Y, profile name must start with SCLxxxxx
- IF SCL type profile created, specific rules apply and will force an error if not followed:
  - (a) Initial Scan SCL user must not be a library, but a valid SCL user in JS300 (the main user in the Production environment)
  - (b) Restore indicator may never be N, always mark with Y
  - (c) Comment must be marked with N, as no text comment will be added to the SCL
  - (d) Allow Source must be Y
  - (e) Source Unload must be N (No SYSOBJH function available for SCL's)
  - (f) Stow/Cat indicator must be S
  - (g) Backup must be marked with Y, backups always made before the move of an SCL
  - (h) ISR Library is not allowed, as SCL's are transferred using the same user, name and type that they were scanned in with.

3.1.2 CC010 – ISR Code

- Code ESPBATCH must be marked with Y for the SCL transfer functions to be available. This will be done at installation time, if a new installation, and can be amended via CC010 after installation if necessary.
- A new code value SCL will be automatically added at installation time for the code ISRTYPE. These 2 codes work together. If at a later stage it is decided not to use SCL transfer anymore, and ESPBATCH is amended to N, then the ISRTYPE SCL must first be deleted, before the amendment can take place.
- Code SCLADHOC will determine if users are allowed to migrate SCL User "ADHOC" directly from Development to Production and vice versa, without going through all the paths or having to create an ISR.
- Code ADHOCL which is used for modules reflects a LIBRARY NAME for an ADHOC Natural Library. This same "ADHOC" name is used in SCLADHOC as the SCL User.

3.1.3 CC330 – Scan SCL's for Transfer

Use this function to scan in SCL's from the EspBatch environment. Only SCL's may be scanned in from this function and objects must be scanned in via CC300 as specified in the manual above. This function is used to populate the EspControl SCL inventory and update the "new-scl" indicator to "old" on all the SCL's being scanned in. The function should **only** be executed during the installation procedure of EspControl. All SCL's scanned will be recorded as existing SCL's in all the environments (to which the SCL has been linked in an ISR to a SCL profile) which means that the ISR transfer flow will **start** from the Master Index environment to the Initial Environment specified on the profile. The SCL scan must take place from the Production environment (and not the Development environment as is the case for normal ISR's).

```
gm712@CRONUS01:/data/analyzer/scripts

CCP330              *** Cronus Consulting - DEV ***              16:15:38.2
CCM330        Scan SCL's for Transfer (Initial Scan ONLY)        2022/09/28
                                                                 GM712




                 Enter SCL User......: CRONUS _
                       SCL Type......: SCL_____
                 Start SCL...........: A______
                 End SCL.............: ZZZZZZZZ
                 Scan environment....: 03 - Production


           Enter * in SCL User to scan ALL Users in SCL selection
           Enter * in SCL Type to scan ALL Types in SCL selection


 Direct Command: _____________________________________________________

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC330 – Scan SCL's for Transfer

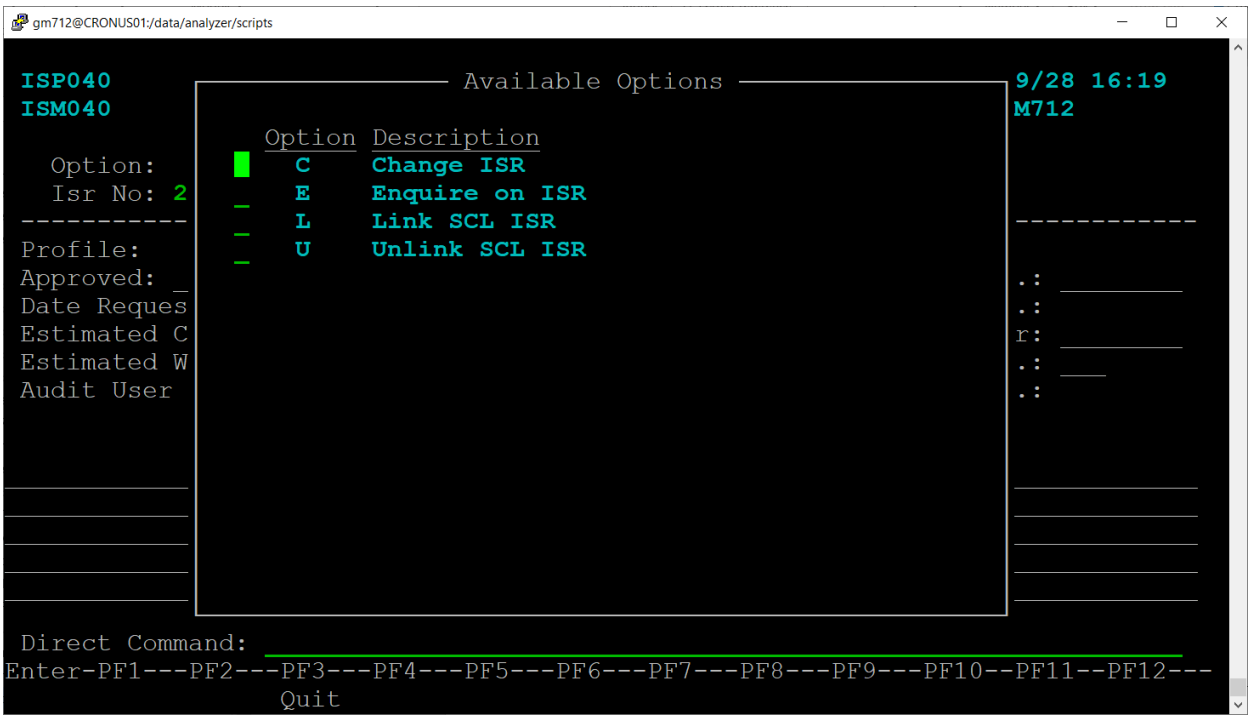
3.1.4 CC350 – Run Initial Backup Routine

This routine must not be executed for SCL's. No initial backup is required, as each SCL will be backed up before the move to the next environment and not after the move as for normal ISR's.

3.2 ISR Transfer Function Overview

3.2.1 CC040 – Approve ISR Request

Approve the SCL ISR as normal. However, the ISRTYPE must be selected as SCL and the profile must have been set up in CC030 as an SCL type Profile. If CC035 is used, then the approval and SCL type will already be entered. To use the link option as described below, even if CC035 was used, CC040 must be used for linking. There is an option where the SCL ISR may be linked to a normal ISR. This will enable the migration to be run in sync and will prompt the user to start the SCL ISR once the normal ISR has completed one move to the next environment. To link the ISR, the normal ISR must be used as the base. In CC040, either C for change the normal ISR, or L for Link of an SCL ISR – but in both cases, use the normal ISR number. Enter the SCL ISR number in the "LINK SCL ISR" field as show in the figure below. This must be a SCL type ISR and must also be in the correct status (new). Once a SCL ISR has been through a migration, it cannot be linked to a normal ISR, unless the entire SCL ISR is restored to the initial environment. Once linked, if the SCL ISR is enquired on, it will reflect the linked normal ISR as enquiry only. An SCL ISR can however be unlinked at any time, or when in any status (unless complete), from the normal ISR. Use option U for the unlinking, and enter the normal ISR number, and then remove the SCL ISR from the linked SCL ISR field.



CC040 – Options L and U

When linking an SCL ISR to a normal ISR, it will only be possible to run the SCL ISR in tandem via CC100 or CC105. If the SCL ISR number is entered in CC100, an error will be given to inform the user to

run the normal ISR. If this is no longer required, unlink via CC040 and then the SCL ISR may again be run individually. If no link of the SCL ISR has taken place, then this SCL ISR may be entered in CC100 or CC105. The linking of an ISR does not affect the Return function CC102. If an SCL of a linked ISR is to be returned, it must be run individually in CC102.

```
gm712@CRONUS01:/data/analyzer/scripts
ISP040              *** Cronus Consulting - DEV ***            09/28 16:34
ISM040                   IT Request Maintenance                GM712

  Option: C
  Isr No: 26________ 01 - Downloaded to Development
 ---------------------------------------------------------------------------
 Profile:  PRD2DEV                          Description:  PROD TO DEV
 Approved: A (A)pproved (R)ejected          Date Received by IT....: 20220920
 Date Request Registered: 20220920          System.................: ACC
 Estimated Comp Date....: 20220920          Development Team Member: GM712___
 Estimated Work Hours...: ___1.00           ISR Type...............: NAT
 Audit User Id..........: GM712             ISR Security...........: N
 Linked SCL ISR.........: 31______
                Description of Change / Other Enhancements   Page: 1 of 6
TEST APPROVAL AND LINKING OF SCL ISR_____________________________________________
_________________________________________________________________________________
_________________________________________________________________________________
_________________________________________________________________________________
_________________________________________________________________________________

 Direct Command: ___________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit                    PgUp  PgDn
```

CC040 – SCL ISR in Linked SCL ISR field

As reflected in the example above, SCL ISR 31 is going to be linked to normal ISR 26. When migrating via CC100 or CC105, only enter ISR number 26, and once a move has completed for 26, the function will automatically call SCL ISR 31 to be migrated as well.

3.2.2 CC050 – Link SCL's to an ISR

When an ISR number is entered and CC050 detects that it is an SCL ISR, it will only display SCL's from the Inventory. SCL's may be linked to an ISR via an 'X' in the select column in the same manner as normal objects. However, no library selection exists as SCL's must be transferred to and from environments with the same SCL user and SCL Type. SCL's may be unlinked by removing the 'X' from the selection column, users will be requested to do a restore (if in correct status) but only current versions restores will be allowed. Previous backup restores are not available as they are in normal object ISR's.

```
gm712@CRONUS01:/data/analyzer/scripts

ISP055                  *** Cronus Consulting - DEV ***                 09/28 16:37
ISM055                     Select SCL's for Transfer                    GM712

 ISR....: 31_________     Status: 00 - Object Selection in Progress
 Profile: SCL2DEV         Desc  : SCL PROD 2 DEV
        User      Name      Type          Current ISR Stat New Scan By     Date
X       DEMO      DAILY     SCL                            Y   GM712    20220928
_       DEMO      DEMOJCL   SCL                            Y   GM712    20220928
_       DEMO      DEMOJCLX  SCL                            Y   GM712    20220928
_       DEMO      DEMOSCL   SCL                            Y   GM712    20220928
X       DEMO      J505S011  SCL                            Y   GM712    20220928
_       DEMO      JGL230    SCL                            Y   GM712    20220928
_       DEMO      JOB1      SCL                            Y   GM712    20220928
_       DEMO      JOB2      SCL                            Y   GM712    20220928
X       DEMO      JOB3      SCL                            Y   GM712    20220928
_       DEMO      JOB4      SCL                            Y   GM712    20220928
_       DEMO      JOB4-1    SCL                            Y   GM712    20220928
_       DEMO      JOB4-2    SCL                            Y   GM712    20220928
                               ** Start of Data **
  Restart at SCL User: ________ SCL Name: ________        No of SCL's:
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
            Del   Quit              Scan        PgDn        Curr
```

CC050 – Select SCL's for Transfer

The only PFKEYS available are PF2, PF3, PF6, PF7, PF8 and PF10. The other PFKEYS visible in CC050 for a normal ISR are not applicable to SCL's and so cannot be selected.

- The common keys PF3 for Quit, PF7 for PgUp, PF8 for PgDn and PF10 for Curr and All operate as normal.
- PF6 – scan in SCL's. If a new SCL has been created in the Development ESPBATCH, scan in this SCL via PF6, which will create a new entry in CC050. 'Y' will be indicated under New and when a migration is done the New SCL will follow the same rules as a New object. There will be no download from Production and because the SCL does not exist, there will be no backup of the SCL in the next environment.
- PF2 – Delete SCL off the Inventory List. Unlike Objects which must be archived via CC250 when removing from the Inventory List in CC050, SCL's are not archived. The actual SCL does not get deleted from EspBatch, only the entry in CC050 gets removed, which means that no further migration will be possible for this SCL, but the SCL will still exist in all environments in EspBatch. Removal from EspBatch must be done via JS300 as is normal for SCL's.

```
gm712@CRONUS01:/data/analyzer/scripts

ISP055                 *** Cronus Consulting - DEV ***                 09/28 16:37
ISM055                    Select SCL's for Transfer                    GM712

 ISR....: 31_________    Status: 01 - Downloaded to Development
 Profile: SCL2DEV        Desc  : SCL PROD 2 DEV
        User                    Scanning SCL's                    can By    Date
X       DEMO                                                      M712    20220928
_       DEMO              Enter SCL's to be scanned               M712    20220928
_       DEMO                   or PF3 to Exit                     M712    20220928
_       DEMO            SCL User : DEMO____  (* for all)          M712    20220928
X       DEMO            SCL Type : SCL_____  (* for all)          M712    20220928
_       DEMO         Start SCL: *_______ - End SCL: ________      M712    20220928
_       DEMO                                                      M712    20220928
_       DEMO            ReNEW Existing SCL : N                    M712    20220928
X       DEMO                                                      M712    20220928
_       DEMO      JOB4      SCL                             Y   GM712    20220928
_       DEMO      JOB4-1    SCL                             Y   GM712    20220928
_       DEMO      JOB4-2    SCL                             Y   GM712    20220928
                             ** Start of Data **
  Restart at SCL User: ________ SCL Name: ________       No of SCL's: 3
 Direct Command: _____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
            Del    Quit              Scan        PgDn        Curr
```

CC050 – New SCL to be scanned

```
gm712@CRONUS01:/data/analyzer/scripts

ISP055                 *** Cronus Consulting - DEV ***                 09/28 16:37
ISM055                    Select SCL's for Transfer                    GM712

 ISR....: 31_________    Status: 01 - Downloaded to Development
 Profile: SCL2DEV        Desc  : SCL PROD 2 DEV
        User                    Archive SCLs                      can By    Date
X       DEMO                                                      M712    20220928
_       DEMO            Enter SCL to be Removed off CC050         M712    20220928
_       DEMO                   or PF3 to Exit                     M712    20220928
_       DEMO               SCL User : DEMO____                    M712    20220928
X       DEMO               SCL Name : SCL_____                    M712    20220928
_       DEMO               SCL Type : JOB4____                    M712    20220928
_       DEMO                                                      M712    20220928
_       DEMO                                                      M712    20220928
X       DEMO      JOB3      SCL                    31    1    Y  GM712    20220928
_       DEMO      JOB4      SCL                             Y   GM712    20220928
_       DEMO      JOB4-1    SCL                             Y   GM712    20220928
_       DEMO      JOB4-2    SCL                             Y   GM712    20220928
                             ** Start of Data **
  Restart at SCL User: ________ SCL Name: ________       No of SCL's: 3
 Direct Command: _____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
            Del    EXIT              Scan        PgDn        Curr
```

CC050 – SCL to be Removed

3.2.3 CC054 – Copy an ISR

The copy function operates as normal but a check exists that will only allow normal ISR's to be copied to normal ISR's and SCL ISR's to be copied to SCL ISR's. I.E. both Copy from and Copy To must be the same type of ISR.

3.2.4 CC100 – Transfer an ISR between Environments

The transfer of SCL's differs to the transfer of normal objects in the manner of the actual transfer, not in the running of the routine CC100. An SCL ISR may be run via CC100 individually or if linked to a normal object ISR, it will be run in tandem with the normal ISR. If the SCL ISR is run individually, CC100 will process as normal with regards to the rules of migration, approvals and returns. The actual transfer of the SCL, whether individually run or linked to a normal ISR, will be detailed below.

Linked ISR's

Only the normal ISR number will be entered and the routine will prompt the user to start the linked SCL ISR, once the normal ISR has completed one migration. In this manner, both the linked normal object ISR and the SCL ISR will always be in the same status and will be transferred together to each environment in the selected profile. Each ISR will use its own profile to determine the migration as an SCL ISR has to be linked to an SCL type profile.

The exception to this rule is if the user is prompted to run the SCL ISR and the user elects to Exit via PF3, and not go through with the SCL ISR. In this case the SCL ISR will be one migration behind the normal object ISR. However, when running CC100 again, even if the SCL ISR is behind the normal ISR, enter the normal ISR number. CC100 will control the migrations of linked ISR's and always ensure that each ISR will be on the same level / status before running the next migration. In this case, when entering the normal ISR number via CC100, CC100 will inform the user that the SCL ISR is one migration behind and immediately transfer the user to the SCL migration and will ignore the normal ISR migration, until the linked SCL ISR's are in the same status. See screen images below to see detail. The rules of CC100 will apply to both linked ISR's and will not allow any migration of either ISR to be commenced until all the validations have been passed. For example, if only the SCL profile specifies approval and when running CC100, the normal ISR does not request approval, the normal ISR will not be processed and an error will be displayed that the linked SCL ISR must first be approved. This will apply to approvals, user / group security, return indicators, ISR's in hold. If a normal ISR goes through the final migration, completes and then the user exits before the completion migration cycle of the SCL, the next time CC100 is run, the SCL ISR number must be used, even though it was linked. The user will be prompted to do this. If at any time, a linked SCL ISR is not required to be linked anymore, use option U in CC040 to unlink the ISR. Note, an SCL ISR can only be linked to a normal ISR if no migration of the normal ISR has taken place.

SCL Transfer

The running of an SCL ISR performs the following tasks (this differs from normal object ISR's).

- EXPORT of SCL from the FROM Environment
- BACKUP of SCL in the TO Environment (before overwriting SCL with new transferred SCL)
- IMPORT of SCL into the TO Environment

Audit Report

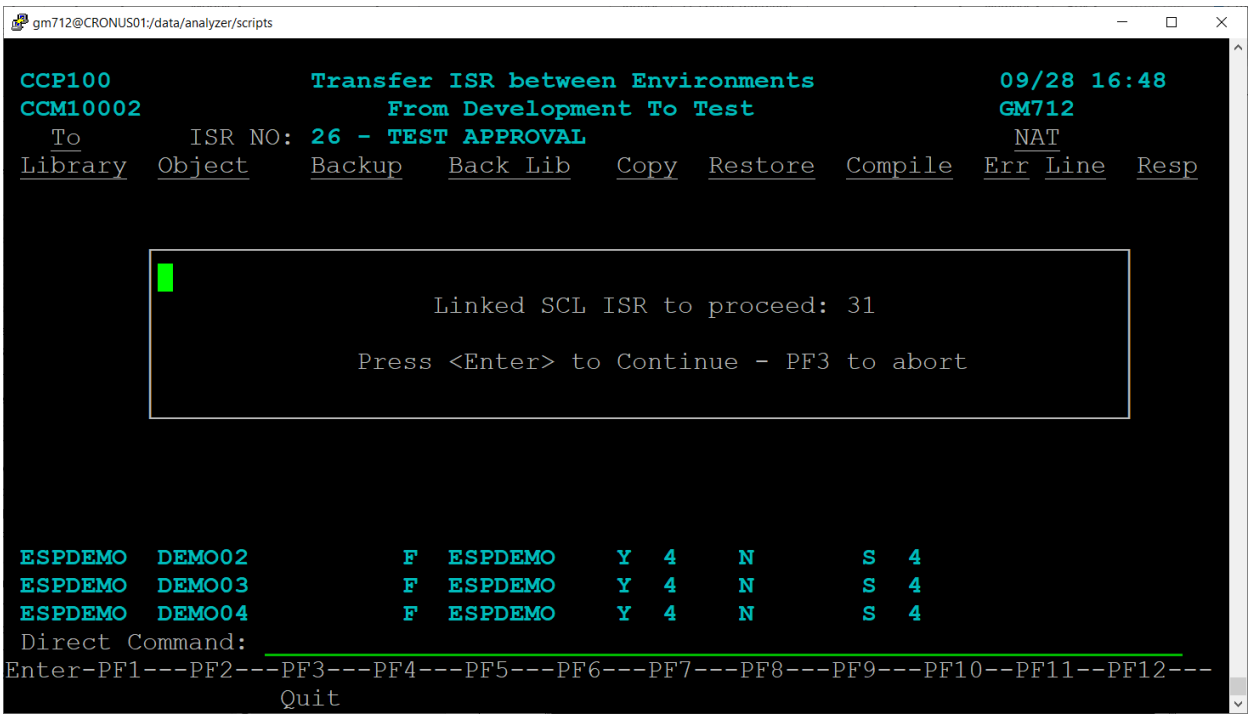
If the code AUDITRPT is set to 'Y' in ISR Codes (CC010), an audit report will be produced in the same manner as the audit report for normal object ISR's. If a SCL ISR has been linked to a normal ISR, two separate reports, one for each ISR number will be produced. The SCL audit report will reflect the SCL indicators Export, Backup and Import and the SCL User, Name and Type, with the rest of the display and format the same as the normal object ISR report.

```
gm712@CRONUS01:/data/analyzer/scripts
Job number 682118: N.20220928.gm712.01 on form type NOPRINT                START
....5....10...15...20...25...30...35...40...45...50...55...60...65...70...75....
2022/09/28                    Audit Report for transfer of ISR No - 31          >
17:01:08.5

ISR Title....: TEST SCL LINK           Xref Number....:                         >
               TEST SCL LINK                                                    >

Environment From: Test
Environment To:   Production

LINE NO   SCL      SCL      SCL    EXPORT BACKUP IMPORT RESTORE
          USER     NAME     TYPE    IND    IND    IND     IND
------- -------- -------- -------- ------ ------ ------ -------

0001              Upload from Test to Production

0002    DEMO     DAILY    SCL      Y      N      Y      N
0003    DEMO     J505S011 SCL      Y      N      Y      N
0004    DEMO     JOB3     SCL      Y      N      Y      N

0005    Uploaded from Test to Production - SCL - BACKUP                         >
                              --- End of file ---
```

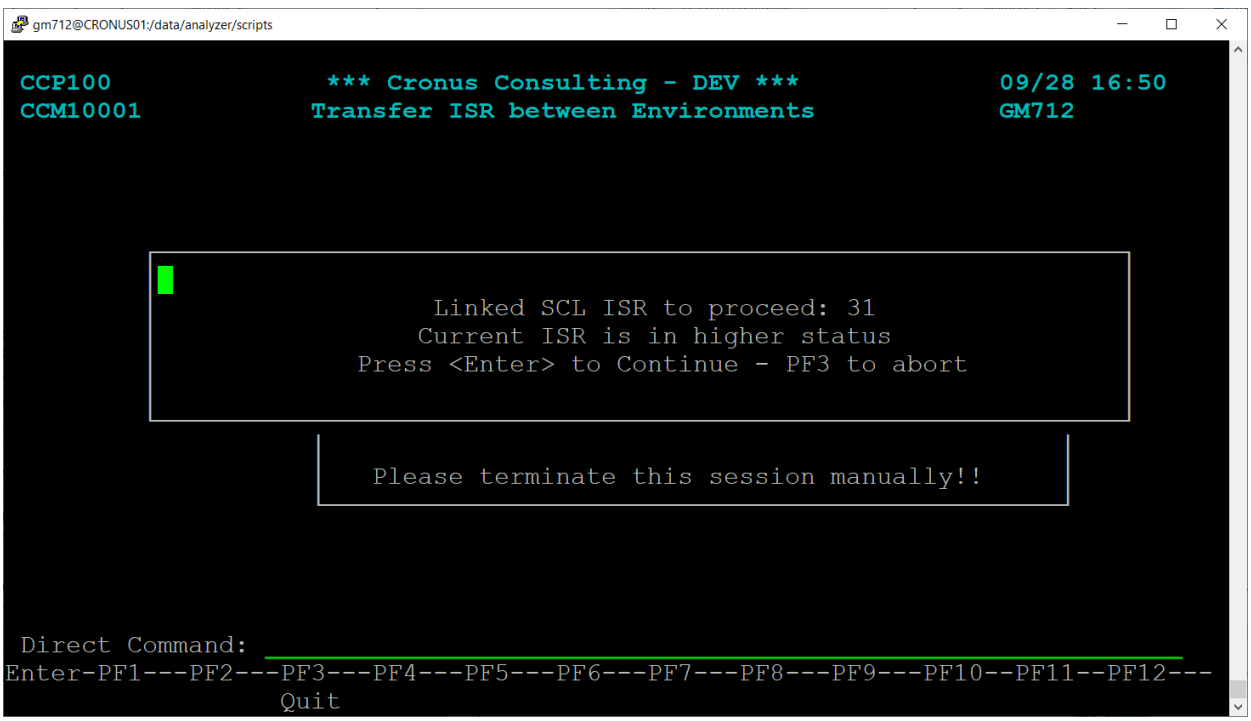
CC100 – Audit Report produced for SCL ISR if AUDITRPT is set to Y

Examples of CC100 for transfer of SCL's:

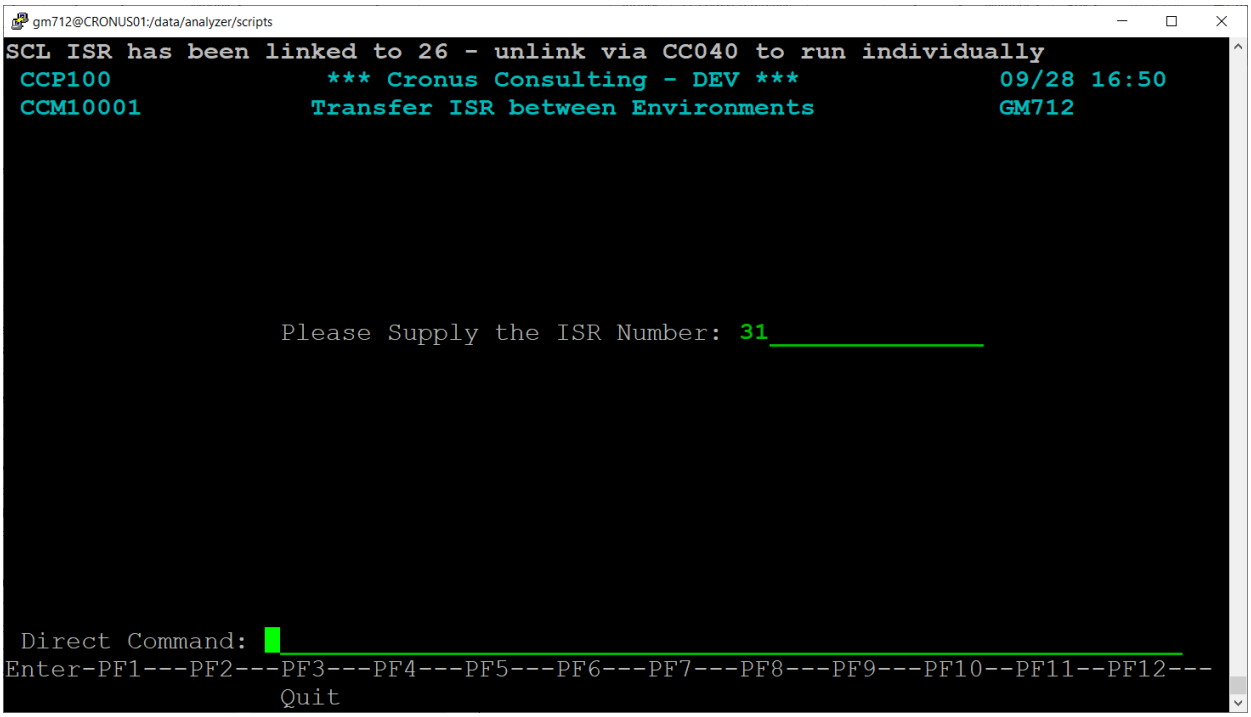
See below for examples of validation rules, linked SCL's, the actual transfer of an SCL, migrations that abort and the display of error messages.



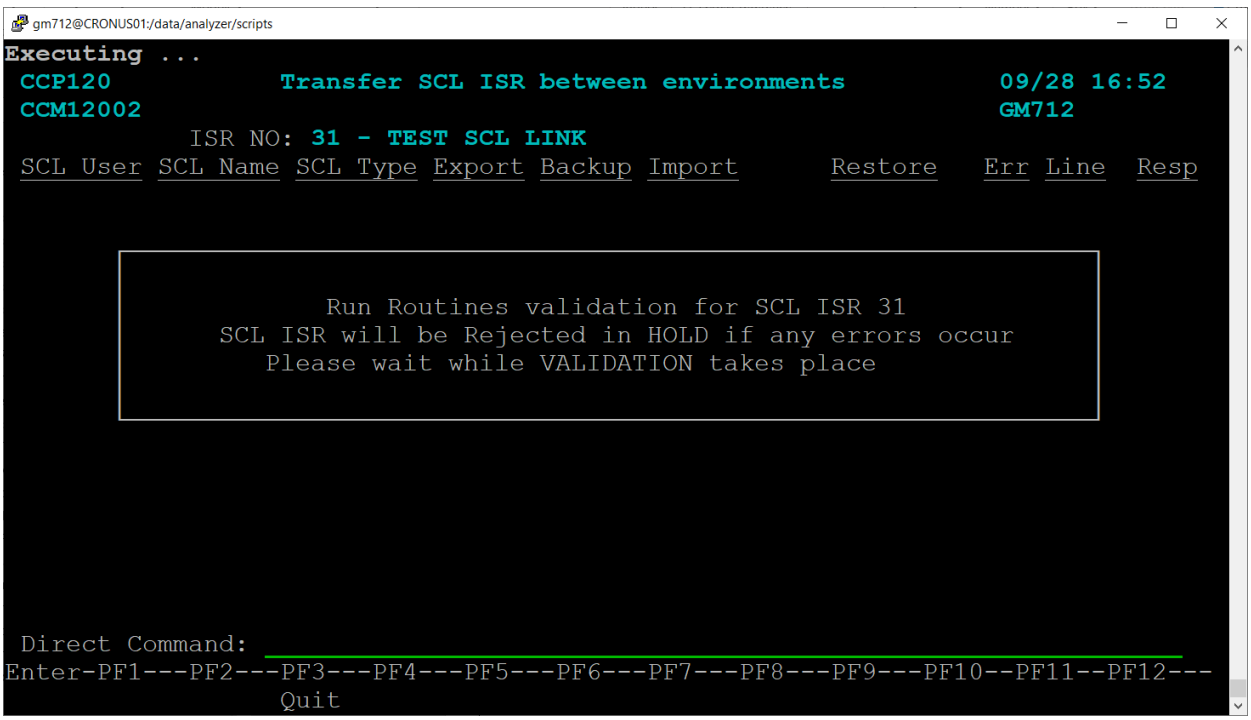
CC100 – Linked SCL prompt



CC100 – Linked SCL in lower status than Normal, Normal ISR bypassed



CC100 – If SCL ISR has been linked, it cannot be run individually



CC100 – Validation routines Window before start of SCL Migration

```
gm712@CRONUS01:/data/analyzer/scripts
Executing ...
 CCP120            Transfer SCL ISR between environments          09/28 16:52
 CCM12002                 From Development To Test                GM712
            ISR NO: 31 - TEST SCL LINK
 SCL User SCL Name SCL Type Export Backup Import      Restore   Err Line  Resp

        Running of Export and Import Routines for SCL's

             Please wait while Routines Complete

 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC100 – Wait Window for SCL's during actual Migration

```
gm712@CRONUS01:/data/analyzer/scripts
Linked SCL ISR not Upload Approved: 31
 CCP100              *** Cronus Consulting - DEV ***             09/28 16:53
 CCM10001           Transfer ISR between Environments            GM712

                 Please Supply the ISR Number: 26

 Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit
```

CC100 – Example that validation rules apply to linked SCL ISR before any migration

```
gm712@CRONUS01:/data/analyzer/scripts
<ENTER> to complete ISR
 ISP070                *** Cronus Consulting - DEV ***            09/28 16:59
 ISM07001                  Mark an ISR as Completed               GM712

   ISR Number: 26_____________
   ISR Status: 03 - Uploaded to Production
-------------------------------------------------------------------------------
   Profile:    PRD2DEV    PROD TO DEV
-------------------------------------------------------------------------------
 ** If Manual Complete and objects should not be restored, Mark with N **
            Restore.........: Y (Y/N/P - Yes/No/Previous ISR)
-------------------------------------------------------------------------------


           Actual Work Hours: __1.00

            Completion Date.: 20220928

            Enter to Continue with Linked SCL ISR


 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
```

CC100 – When Normal ISR completes, user will be prompted to start linked ISR

SCL Transfer

See below figure for completion window of an SCL Transfer. Notice the different indicators for an SCL transfer viz. Export, Backup and Import, the 3 processes for migration of an SCL. The restore indicator is also displayed and if N, no errors occurred and the SCL will complete successfully.

A SCL transfer will export all SCL's, then Backup all SCL's and then Import all SCL's. If a problem occurs in any of the 3 processes, the ISR will abort on the error SCL and only restore the SCL's already through the specific processes. Therefore, no restore is done if the Export process aborts as no update has taken place to the actual SCL. Backups are created before the move to an environment and are also linked to a version number per ISR. See function CC200 in the Appendix below for detailed explanation of how the backup version is created and linked to an ISR.

```
gm712@CRONUS01:/data/analyzer/scripts
CCP120          Transfer SCL ISR between environments          09/28 17:01
CCM12002                From Test To Production                GM712
           ISR NO: 31 - TEST SCL LINK
SCL User SCL Name SCL Type Export Backup Import      Restore   Err Line  Resp

                Uploaded from Test to Production

                 Press <Enter> to Continue

DEMO     DAILY    SCL       Y      N      Y           N
DEMO     J505S011 SCL       Y      N      Y           N
DEMO     JOB3     SCL       Y      N      Y           N
Direct Command:
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit
```

CC100 – Completion Window showing indicators for Export, Backup and Import

3.2.5 CC105 – Transfer with Skip of Environment

CC105, transfer with the skipping of an environment, works in the same manner as CC100 for SCL's, except that an environment will be skipped and the SCL will be transferred to an environment plus one. (As is done in CC105 for normal objects, see further detail in CC105 above in manual).

Individual SCL ISR's may be run via CC105 on their own, and linked SCL ISR's may be run via the normal object ISR as is explained in CC100 for SCL's above. The only exception being, that the ISR's must be in synch, status wise, else it will not allow the skip.

3.2.6 CC250 – Select Objects for Archiving

This routine must not be executed for SCL's. NO SCL's are archived. If it is required to remove an SCL from the Inventory List, use PF2 in CC050 on a valid SCL ISR, to remove the SCL from CC050. This is explained in CC050 in the SCL section above.

3.3 ISR Maintenance Function Overview

3.3.1 CC051 – Transfer a SCL to another ISR

The transfer function operates as normal object ISR but a check exists that will only allow objects from normal ISR's to be transferred to another normal ISR and SCL's from SCL ISR's to be transferred to another SCL ISR. I.E. both Transfer from and Transfer To must be the same type of ISR.

3.3.2 CC076 – Remove ISR from SYSOBJH Error

This function will not be used for SCL type ISR's, as an SCL cannot use the SYSOBJH function. An error will be returned if a SCL type ISR is entered via CC076.

3.3.3 CC077 – Remove ISR from System Message Error

This function will not be used for SCL type ISR's. System error messages must be transferred via normal ISR's. An error will be returned if a SCL type ISR is entered via CC077.

3.4 Display ISR Information

The functions specified below, have all been amended in the same manner. There is an indicator which switches between Objects and SCL's and objects will be reflected when on the Objects screen and all SCL's will be reflected when on the SCL screen. Screen prints will be shown in the function description below (CC085), but all other functions specified under 3.4 Display ISR Information, will have the same set of indicators and operate in the same manner as CC085. The initial screen displayed will always be for Objects. To view SCL's mark Y in the SCL indicator. This will transfer you to the SCL view, and again, to go back to the Object view, mark Y in the Object indicator. Amend SCL user (in field Restart at SCL User) to view other SCL Users. The default SCL user in the display will be taken from the last updated SCL profile.

3.4.1 CC085 – Display Object Dependents

```
gm712@CRONUS01:/data/analyzer/scripts

ISP0851              *** Cronus Consulting - DEV ***              09/28 18:31
ISM08502            Display SCL Detail - No Dependants            GM712
                                   Current
  SCL User SCL Name SCL Type         ISR      Stat New Scan By   Date
  DEMO     DAILY    SCL                                N  GM712    20220928
  DEMO     DEMOJCL  SCL                                Y  GM712    20220928
  DEMO     DEMOJCLX SCL                                Y  GM712    20220928
  DEMO     DEMOSCL  SCL                                Y  GM712    20220928
  DEMO     J505S011 SCL                                N  GM712    20220928
  DEMO     JGL230   SCL                                Y  GM712    20220928
  DEMO     JOB1     SCL                                Y  GM712    20220928
  DEMO     JOB2     SCL                                Y  GM712    20220928
  DEMO     JOB3     SCL                                N  GM712    20220928
  DEMO     JOB4     SCL                                Y  GM712    20220928
  DEMO     JOB4-1   SCL                                Y  GM712    20220928
  DEMO     JOB4-2   SCL                                Y  GM712    20220928
  DEMO     JOB4-3   SCL                                Y  GM712    20220928
  DEMO     PRED1    SCL                                Y  GM712    20220928

  Restart at SCL User: _______ SCL Name: ________    Objects: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit                     PgUp  PgDn
```

CC085 – SCL indicator was marked with Y

3.4.2 CC088 – Display Object History

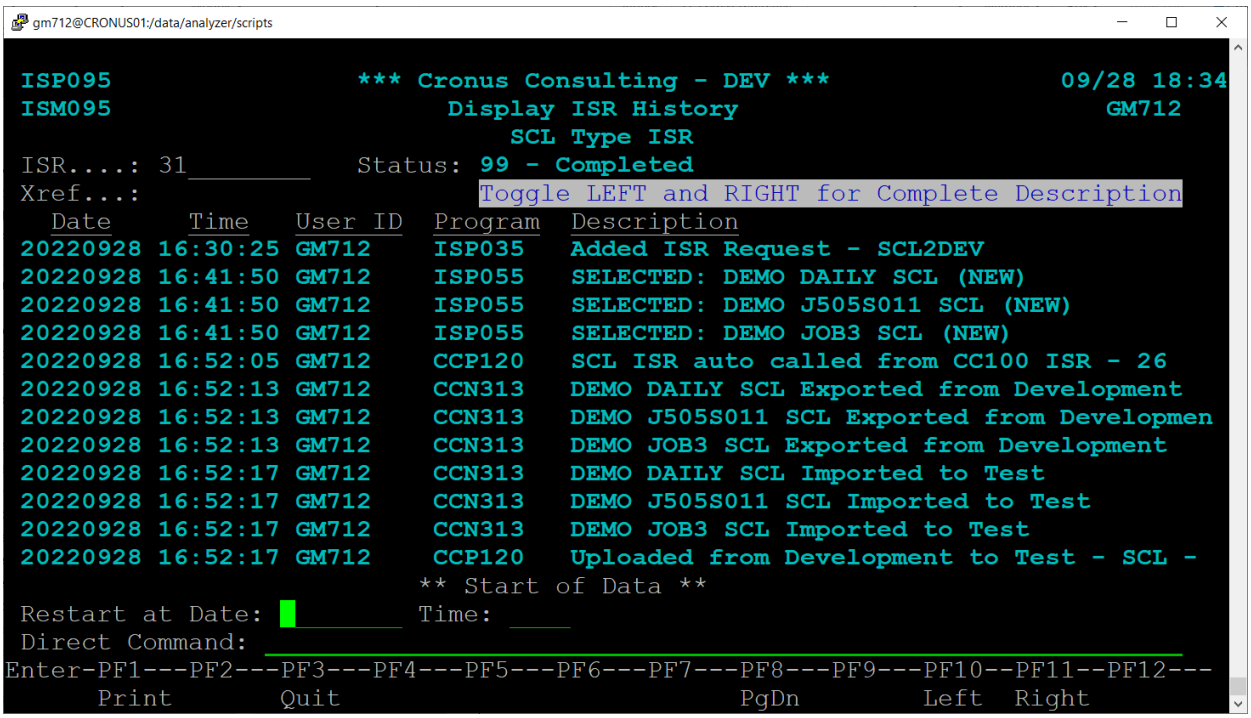
As shown above in CC085, mark the SCL or Object indicator to switch between the different screens.

3.4.3 CC090 – Display Objects linked to an ISR

Depending on the ISR number entered, the type will be picked up and either Objects (with Object Name and Library) will be reflected or SCL's (with SCL User, Name and Type) will be reflected.

3.4.4 CC095 – Display ISR History

Both SCL type ISR's and normal Object ISR's use the same enquiry. There is no difference to the actual enquiry, except the details displayed. The major difference is that SCL's are Exported, Backed Up (before the migration to any environment) and Imported in each move. The Objects are Backed Up (after the migration to all non-production environments and before the migration to Production), Copied and Stowed in each move.



CC095 – History Detail for SCL ISR – Export, Backup and Import

PF5 for Audit History can be run for both Objects and SCL's. Mark either OBJECT or the SCL indicator with Y, not both.

3.4.5 CC088 – Display Object Status

As shown above in CC085, mark the SCL or Object indicator to switch between the different screens.

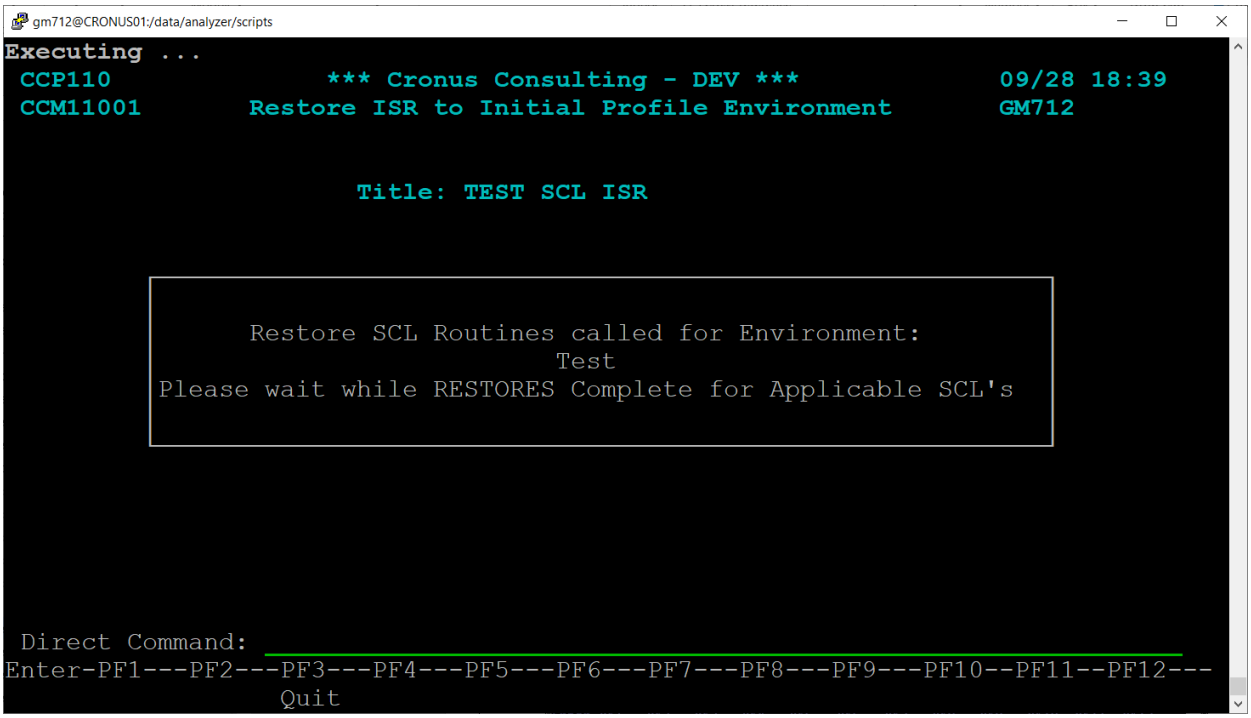
3.5 ISR Return And Restore Function Overview

3.5.1 CC102 – Return Transfer ISR to Previous Environment

The return of an SCL works in the same manner as explained above in the SCL appendix for CC100, escape the SCL will be returned to the previous environment. The only difference is that a linked SCL ISR must be run individually in CC102 and will not be called automatically as in the case of CC100 linked ISR's. SCL's that are to be returned must be updated via CC053 in the same manner as objects are and must also be approved via CC061 (unless RETAPP code is set to N).

3.5.2 CC110 – Restore ISR to Initial Profile Environment

Operates in the same manner as objects, but because the backup cycle is different to that of objects, when restoring, it will always restore to the version before the environment move. SCL's are restored all at once in each applicable environment, whereas objects are done one at a time in all environments. See screen image below as these will differ from the restore screens of normal object ISR's.



CC110 – Restores all SCL's Per Environment

3.5.3 CC115 – Restore Object(s) to Initial Profile Environment

Operates in the same manner as objects, where applicable SCL's must be selected for restore and not the entire ISR as in CC110. Once the SCL's have been selected, the restores will complete as in CC110 but only for the selected SCL's. Screen images will be the same as in CC110.

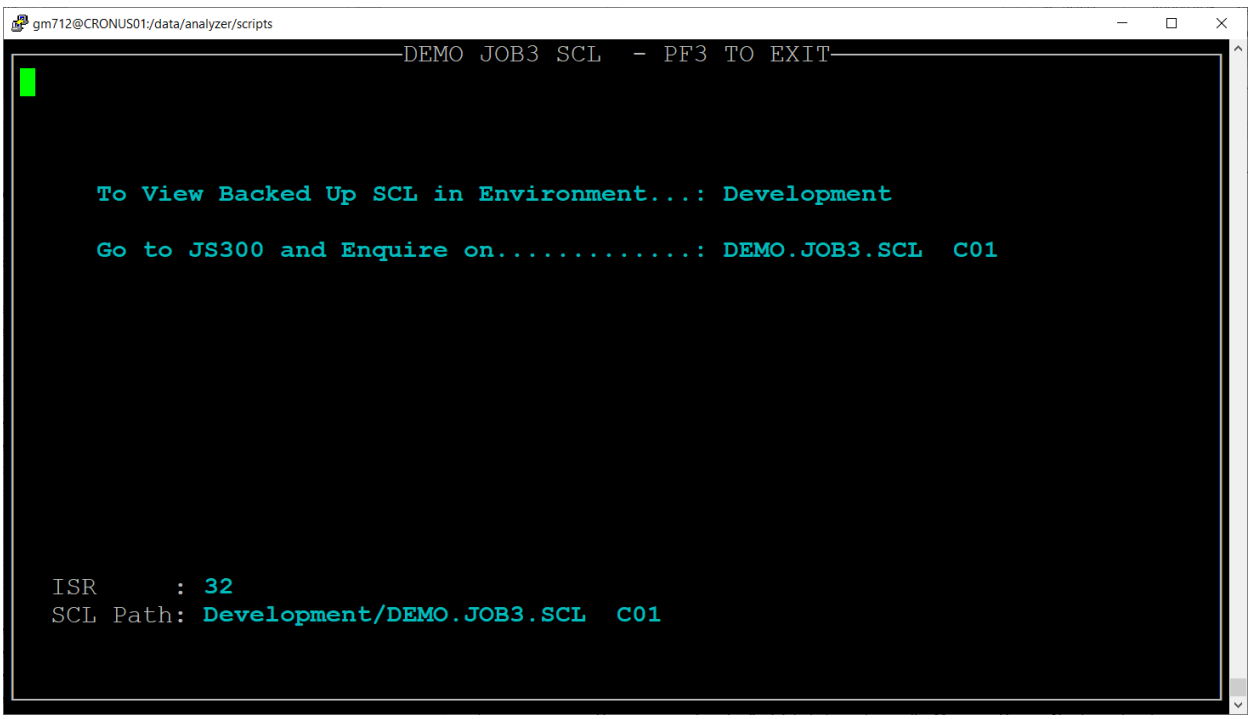
3.5.4 CC200 – Restore Object Versions to ANY Env per ISR

As shown above in CC085, mark the SCL or Object indicator to switch between the different types of ISR's. Depending on your selection, either all objects or all SCL's will be displayed. Amend SCL user (in Restart at SCL User field) to view other SCL Users. The default SCL user in the display will be taken from the last updated SCL profile.

Both the selection process for the applicable SCL's and the type of either D for display or R for Restore, and the environments, work as in the normal Object CC200. However, the actual Display and Restore functions differ for SCL's.

DISPLAY:

A backed up SCL is actually created in EspBatch and may be viewed in JS300 in the applicable environment. For this reason, the actual source of the backup will not be displayed in the Display function, but only the environment and backup version number for the user to view in JS300. This version number will be different for each ISR, in the same manner as objects have version numbers per ISR.



CC200 – Display function showing where to view applicable backup

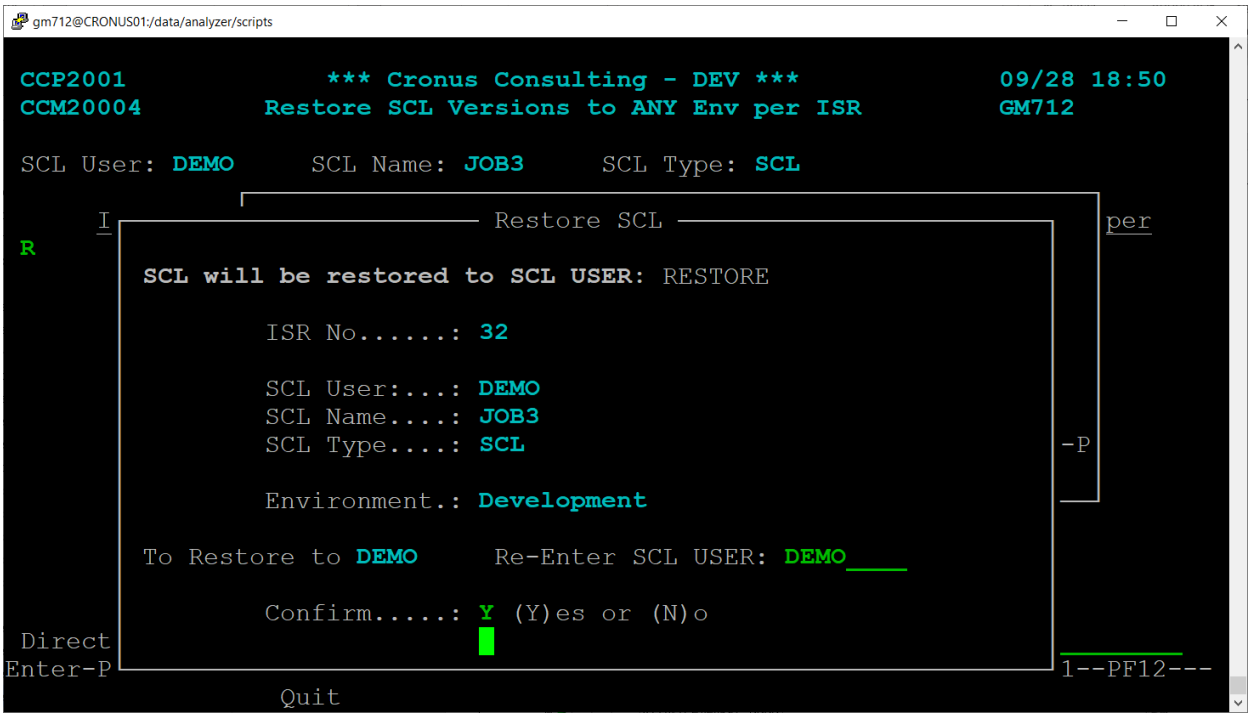
RESTORE:

The restore function for SCL' s will be restored to SCL User RESTORE by default, in each of the selected environments. If the SCL must be restored to the original SCL User, this user must be entered in the field as seen below (Re-Enter SCL User). If the field is left blank the original SCL will not be affected, but the applicable backed up version will be restored to SCL User – RESTORE. The restore function will restore from the linked backup version to the selected ISR, but the restored version will not include the version number Cnn in the SCL Type, it will restore to the original SCL Type that excludes the Cnn.

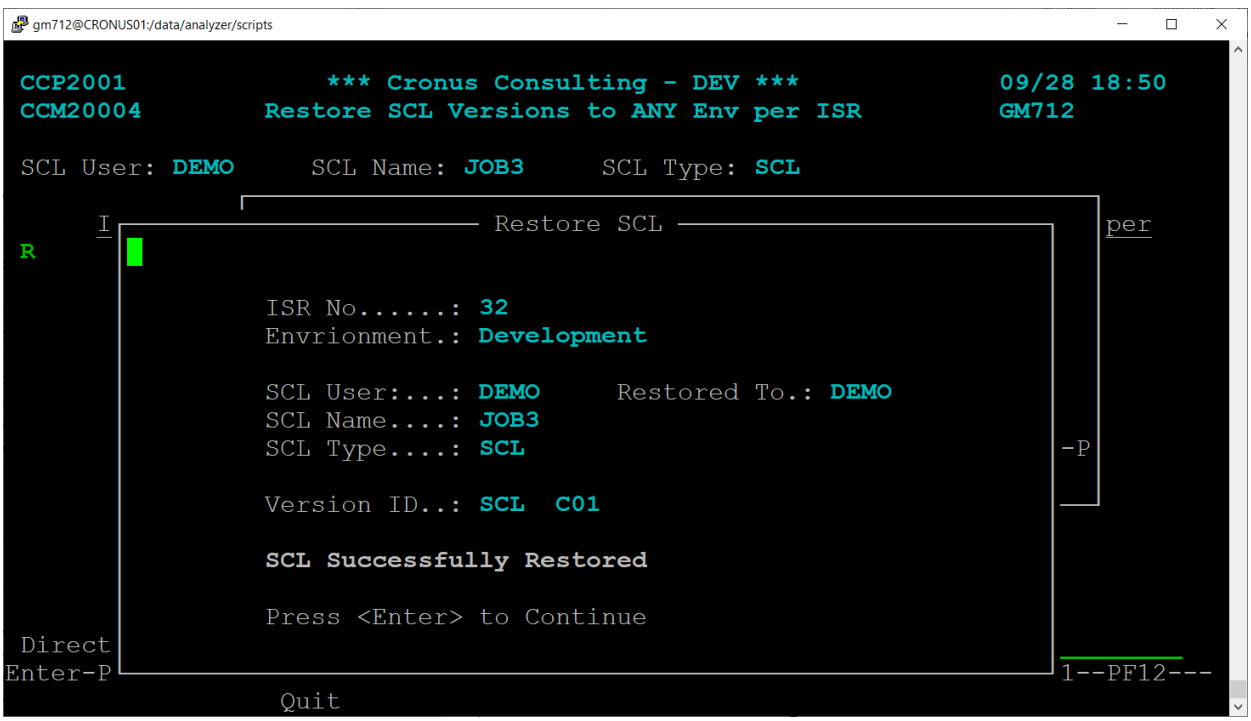
Explanation of Backed Up Versions:

- Whenever a migration takes place, the SCL is backed up before the import to the next environment.
- A version number is allocated per ISR (same version number for each environment)
- The backed up version is actually stored in EspBatch in the same file as the normal SCL's.
- The backed up version may be viewed in JS300 (Enquiry and Print only), it may not be amended or deleted.
- The SCL Type of each backed up version is created as follows - SCL Type "C" version number, for e.g. SCL C03. This backup version is created in the same manner the "Versioning" Option in JS300 which a user may create. The "Versioning" option creates a "V01", whereas the ISR backup option uses a "C01" etc.
- Versions are automatically deleted, depending on the Maximum Version Number entered in CC001. Once the maximum number of versions has been reached the oldest version will be deleted. This will mean that although a link may exist to a backup version number, it will no longer exist and any restore or display in CC200 will specify that the backup cannot be found.

See examples below of Restore Functions.



CC200 – Restore function to restore to Original User



CC200 – Successful Restore showing Version used for Restore

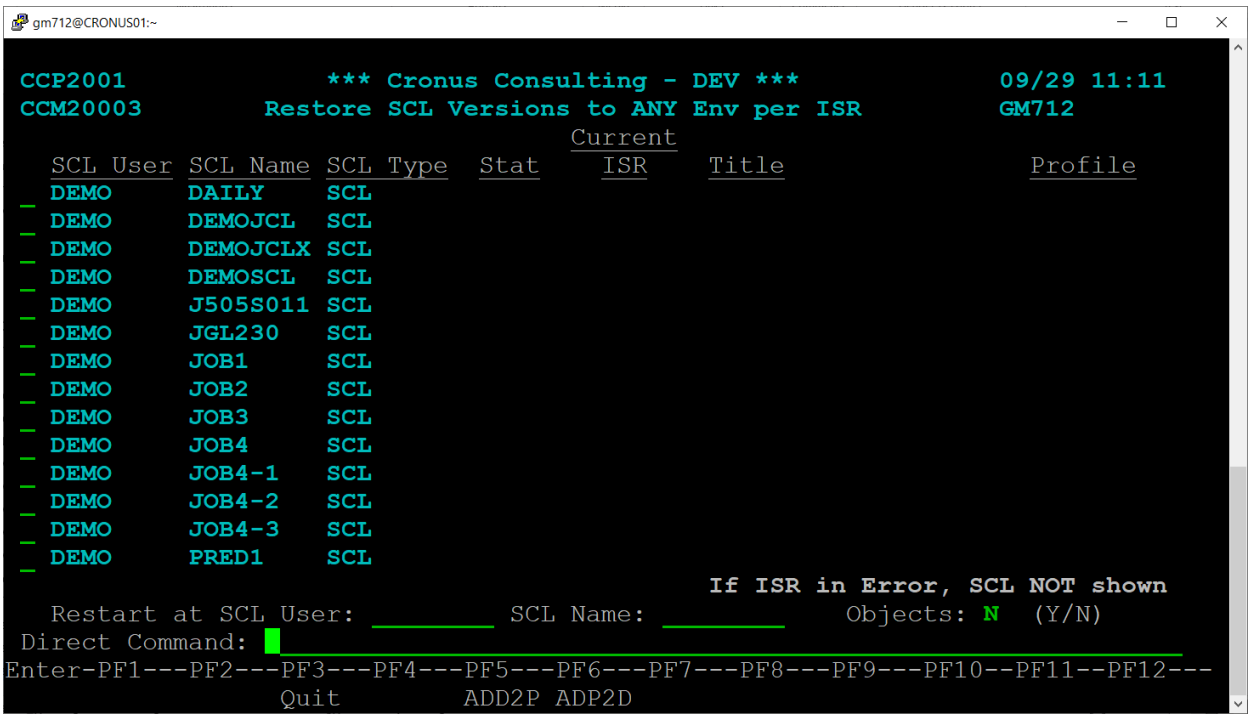
Option ADHOC - Code SCLADHOC is used to determine if SCL's can be moved from SCL User ADHOC to either Production SCL User ADHOC or Development SCL User ADHOC. The ADHOC SCL User is determined by the code ADHOCL in the same manner as it determines the ADHOC library. PF5 is used for SCL ADHOC copies from Development to Production. PF6 is used for SCL ADHOC copies from Production to Development.

The SCLADHOC function will only be allowed if the 1st code = 'E' or "Y'. If it is set to N, then PF5 and PF6 will not be allowed. The 2nd code is the Production environment number and the 3rd code is the Development environment number. (See detailed explanation in CC010).

If the 1st code is set to 'E', then if code SCLADHOC has a set of users, not only the default user ADHUSER, the SCL ADHOC function will be limited to this set of users.

The SCL ADHOC function will use the environment numbers in code ADHOCL to determine the development and production environment.

CC095 keeps a simple log of all ADHOC copies (both object and SCL) and can be seen when selecting PF6 which is always linked to ISR 77777777.



PF5 and PF6 for the SCL ADHOC function

```
gm712@CRONUS01:~
CCP2001              *** Cronus Consulting - DEV ***              09/29 11:11
CCM20003         Restore SCL Versions to ANY Env per ISR          GM712
                                      Current
  SCL User SCL Name SCL Type  Stat     ISR     Title                Profile
_ DEMO     DAILY    SCL
_ DEMO              Enter ADHOC SCL
_ DEMO
_ DEMO                SCL User: ADHOC
_ DEMO          Enter SCL Name: TEST____
_ DEMO          Enter SCL Type: SCL_____
_ DEMO          To be copied FROM Dev
_ DEMO                TO PRODUCTION
_ DEMO
_ DEMO                PF3 to Exit
_ DEMO
_ DEMO     JOB4-2   SCL
_ DEMO     JOB4-3   SCL
_ DEMO     PRED1    SCL
                                          If ISR in Error, SCL NOT shown
  Restart at SCL User: ________ SCL Name: ________    Objects: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit        ADD2P ADP2D
```

PF5 copy from Development to Production – enter the SCL Name and Type

```
gm712@CRONUS01:~
CCP2001              *** Cronus Consulting - DEV ***              09/29 11:11
CCM20003         Restore SCL Versions to ANY Env per ISR          GM712
                                      Current
  SCL User SCL Name SCL Type  Stat     ISR     Title                Profile
_ DEMO     DAILY    SCL
_ DEMO              SCL ADHOC Transfer
_ DEMO
_ DEMO                SCL: ADHOC.TEST.SCL
_ DEMO
_ DEMO                could not be copied from DEV
_ DEMO                to Prod ADHOC     - Error
_ DEMO                SCL Not Found for Exporting
_ DEMO
_ DEMO                Press <Enter> to Continue
_ DEMO
_ DEMO
_ DEMO     JOB4-3   SCL
_ DEMO     PRED1    SCL
                                          If ISR in Error, SCL NOT shown
  Restart at SCL User: ________ SCL Name: ________    Objects: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                 Quit        ADD2P ADP2D
```

PF5 SCL ADHOC copy error

```
gm712@CRONUS01:~
CCP2001              *** Cronus Consulting - DEV ***              09/29 11:17
CCM20003         Restore SCL Versions to ANY Env per ISR          GM712
                                      Current
  SCL User SCL Name SCL Type  Stat     ISR     Title                Profile
_ DEMO     DAILY    SCL
_ DEMO              Enter ADHOC SCL
_ DEMO
_ DEMO                SCL User: ADHOC
_ DEMO          Enter SCL Name: TEST1___
_ DEMO          Enter SCL Type: SCL_____
_ DEMO          To be copied FROM Prod
_ DEMO                TO DEVELOPMENT
_ DEMO
_ DEMO                PF3 to Exit
_ DEMO
_ DEMO     JOB4-2   SCL
_ DEMO     JOB4-3   SCL
_ DEMO     PRED1    SCL
                                          If ISR in Error, SCL NOT shown
  Restart at SCL User: ________ SCL Name: ________    Objects: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        ADD2P ADP2D
```

PF6 copy from Production to Development – enter the SCL Name and Type

```
gm712@CRONUS01:~
CCP2001              *** Cronus Consulting - DEV ***              09/29 11:17
CCM20003         Restore SCL Versions to ANY Env per ISR          GM712
                                      Current
  SCL User SCL Name SCL Type  Stat     ISR     Title                Profile
_ DEMO     DAILY    SCL
_ DEMO              SCL ADHOC Transfer
_ DEMO
_ DEMO                SCL: ADHOC.TEST1.SCL
_ DEMO
_ DEMO                has been copied from PROD
_ DEMO                to DEV ADHOC successfully
_ DEMO
_ DEMO                Press <Enter> to Continue
_ DEMO
_ DEMO
_ DEMO
_ DEMO     JOB4-3   SCL
_ DEMO     PRED1    SCL
                                          If ISR in Error, SCL NOT shown
  Restart at SCL User: ________ SCL Name: ________    Objects: N  (Y/N)
 Direct Command: ____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        ADD2P ADP2D
```

SCL – ADHOC.TES1.SCL successfully copied to SCL User ADHOC in Development