

EspControl

Quick Reference Guide





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	4
1.1 PURPOSE	4
1.2 PRE-REQUISITES	4
1.3 PRODUCT OVERVIEW	5
2. MENU FUNCTIONS	7
3. ORDER OF EVENTS.....	8
3.1 INSTALL	8
3.2 SETTING UP THE ENVIRONMENT VARIABLES – CC001 (CC500)	8
3.3 CREATION OF THE PROFILES REQUIRED – CC002 (CC500)	10
3.4 SCAN IN SOURCE CODE – CC300 (CC500)	10
3.5 INITIAL BACKUP ROUTINE – CC350 (CC500)	11
3.6 CREATE AND APPROVE AN ISR – CC030 (CC501)	12
3.7 LINK OBJECTS TO AN ISR – CC050 (CC501)	13
3.8 TRANSFER AN ISR – CC100 (CC501)	15
3.9 COMPLETE AN ISR – CC070 (CC502)	17
4. EXTRA ROUTINES	18
4.1 ARCHIVING – CC250 (CC501)	18
4.2 RESTORES – CC200 (CC504)	18
4.3 ISR QUERY FUNCTIONS - CC503	20



1. Introduction

This Quick Reference guide is to be used in conjunction with the EspControl User Manual for more detail and instructions on the actual execution and rules of the specified functions. If otherwise mentioned, see the manual referred to. If the user has specific queries regarding the functions in EspControl User Manual, scan for these functions using the option number to get related detail e.g. CC050. Many functions are incorporated into the summary headings below.

1.1 Purpose

EspControl has been designed to assist the system administrator in the migration, documentation and backup of Natural Objects between environments on a Unix/Linux platform.

Correct usage of the change control system, will ensure that any given module 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 user to migrate ISR's to and from master environment.

1.2 Pre-Requisites

- Depending on the SHELL environment selected, 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.

If EspBatch is also installed - see Appendix A in the full EspControl Manual for detail explanation of SCL transfer. All functions operate as explained below, with the detail differing for SCL's.



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
- Multiple backups of Source and Object Code 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 environment 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.
- Dependencies allow an object and all its dependents to be selected (if the allow source option is valid and source exists in Development)
- Automatic restore capability
- Back out and restore on migrated objects
- 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.
- 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.
- Libraries may be set up using a GENERIC library in the Profile. Together with using GENERIC, the user exit ISNUX100 must be amended to insert the formula that will translate GENERIC into the correct library used.



-
- User or Group security per Profile.
 - ISR Security where only the “owner” may approve, link objects and upload approve an ISR.
 - 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.
 - Source can be unloaded via the SYSOBJH function per environment creating a UNIX 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 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.
 - If Current Environment includes EspBatch, then SCL's may also be migrated from one environment to the next. All required routines will have an option for SCL's. These will only be available if EspBatch is running.



2. Menu Functions

The Change Control System consists of six parts and is reflected on a main-menu with sub-menu's linked, with each reflecting a subset of functions, which will be displayed to each user depending on the security of the allocated function. There are five sub-menu's and each is grouped together with functions that work in conjunction with each other.

- Control Variables and Profiles – CC500
- Initial Backup Routine – CC500
- ISR Transfer – CC501
- ISR Maintenance – CC502
- Display ISR Information – CC503
- ISR Return and Restore – CC504

See main menu below.

```

Cronus Consulting
File Edit Options Send Receive Window Help

MAP902          *** Cronus Consulting *** V7.1.2          13:33:53.0
MAM902          - CC : EspControl - Change Control System (I.S.R) 2012/02/17

CC500          + ISR Control and Profiles
CC501          + ISR Transfer
CC502          + ISR Maintenance
CC503          + Display ISR Information
CC504          + ISR Return and Restore

*** End of Data ***

Printer.: PRT01                               User: MENUADM
Function: _____ Data: _____          Term: pts/3
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
          Quit                               PgDn SetPr          Logof

VT220 TCP/IP 13:30 0000

```

Figure 1: Main Menu

See EspControl User Manual for detail of each sub-menu. The functions mentioned in the Order of Events below may be used in the function command line above, without actually going into the sub-menu linked to that function. Alternatively, enter each sub-menu specified above and select the options on that particular menu. The function will be specified in **blue** and the linked sub-menu will be added in brackets in **red**, in the explanations below.



3. Order Of Events

3.1 Install

- Install EspControl (see Installation Manual)
- Ensure the temporary directory \$ESPCCTMP (this is set up as \$CRONUS/espccctmp initially) is set up with the correct permissions as the Change Control scripts use this directory for trace files. Ensure that this variable is correctly populated for each user as well, for e.g. set up in the sysenv.setup in each environment or at login time.
- With the running of the main install program (INST-CC), the main menu functions, menu layout and ISR related Codes are created
 - i) **MA** - For checking Menu functions and Administration and linking of users, use menu administration menu MA (See EspMenu Manual)
- **CC010 (CC500)** - For ISR Codes – these codes run the setup of the ISR routines and new codes may NOT be added, but values to the existing codes may be amended or added to. Each code has specific controlling options and the help routines will display what is valid. ENTER on the specific field for help. If the captured data is incorrect, the help routine will be automatically called and the valid options displayed. Specific defaults are set up in the install program and must be changed before continuing to set up EspControl to suit the user's needs.

3.2 Setting up the Environment Variables – **CC001 (CC500)**

- CC001 - Create the control and environment variables necessary for the site
 - i) Choose the correct remote access shell command for the migrating, backup and restore of objects and ensure that this command is enabled correctly (with correct authorisations) before setting up the actual variables (rsh, ssh, remsh)
 - ii) CC001 – set up control variables necessary for the running of the Change Control
 - iii) Specify all environments that may be used in any migration, although not all the environments need to be used in an actual migration routine. Name these environments and specify all the paths (fuser, backup and source staging areas, file numbers, database ids, buffer pool, natural command etc) that will control the migration. If any of these variables are incorrect, the migration will not complete successfully.
 - iv) Ensure that your path directories exist. Ensure that the directories have the correct permissions.
 - v) Ensure that your entries are the correct case (lower or upper), as if they are not, again the migration will not be successful.
 - vi) Decide how many backups per object will be kept and enter the maximum number of backups. Each ISR and linked object will have the same backup version number per environment. Once the maximum number is reached, the earliest backup will be deleted before the backup routine continues. If the Initial Backup Routine is to be run, add one on to this number.



- vii) The source staging area keeps objects in the environments that have object code only or code that must be removed after a migration. Even if the site does not use object code only or move source, the staging area always keeps a copy of the objects, in case the environment parameters are ever changed. If the site has the same library names per environment, then this staging area **MUST** be different if the environments exist in the same fuser. A code SAMESTAGE is linked and the routine will verify if the same source staging area is allowed. Ensure that this code is set correctly before entering the staging area (CC010).
- viii) Enter the correct shell script path – all the scripts that run the migration tool will exist in this area. Again the migration will not be successful if this area is not correct and also if this area does not contain all the change control scripts. Ensure the scripts all have “execute” permission. For Development the shell script path will default to \$CRONUS/ccont_eco during install time.
- ix) If logs of the migration are required, set CREATE LOG FILE to Y. This is almost essential to do, else if the log file is not set up, debugging unsuccessful migrations will not be possible. This will create a log file (in the shell script path) that is linked to an ISR number. All the movements and errors will be recorded here. The log file is always created in the Development environment (Environment no 1 specified in CC001) for all migration, backup and restore moves, irrespective of the migration environment involved. The error explanation may not be sufficient, for this detail see point below.
- x) An automatic trace file of all commands, via the EspControl scripts, used is also created. This will enable you to determine the problem correctly if a migration does not complete. There are many permissions etc per environment that need to be correct and this log file will reveal what the exact problem is. Go to **cd \$ESPCCTMP** to view these logs. These trace files are removed if the migration is successful, so will only be available on error.
- xi) Once the user has entered an environment and exited from CC001, when next CC001 is used, PF2 must be used if more environments are to be added. Same rules apply as above, but an environment may be copied and amended accordingly. This environment may be inserted in any position.
- xii) As these control variables specify rules for all the migration routines, the user has to complete the capturing of the header record and at least one environment before being allowed to exit, or CLEAR all captured details and then exit and restart later.
- xiii) If new environments are to be added, even after the change control has been in use, insert these using PF2 (same as point xi above).
- xiv) This will also insert a NIL path into all profiles as well, so that all profiles and history will be correct. The ENVIRON code is used to allocate a unique environment number to each environment, even after inserts (CC010).



3.3 Creation of the Profiles required – CC002 (CC500)

- The role of the profile is to determine what rules and what paths a particular migration will follow.
- The CC001 routine will automatically create a profile with the name of MASTER. This MASTER profile is set up using specific defaults and will link all environments as a required path. This profile cannot be changed and is merely a guide as to how to set up other profiles
- The MASTER profile will be automatically amended if CC001 is amended.
- Each created profile will always reflect all environments, but only those whose PATH is marked as Y will be used in transfers.
- The profile always uses the control variables of the specific linked environment in CC001. Therefore if an environment variable is amended, CC002 will always reflect the latest detail.
 - i) Allocate the profile name and a meaningful description.
 - ii) The profile name BACK* is reserved for an initial backup profile.
 - iii) See EspControl User Manual for indicator descriptions and validation rules.
 - iv) Paths marked Y will form the base of your migration and each environment marked Y will be migrated to.
 - v) Therefore, many profiles can be set up to fit in with any migration plan. Mini-paths may also be set up, where the initial environment is not necessarily the first environment i.e. Development.
 - vi) The MASTER environment on the PROFILE refers to the LAST environment that this specific profile will end at.
 - vii) The INITIAL environment refers to the FIRST environment that this specific profile will start with.
 - viii) A profile may be set to either DOWNLOAD or UPLOAD, if the PATH-CHOICE is set to Y, or UPLOAD ONLY if set to U. In DOWNLOAD, the objects will first be moved from the MASTER and then back up to the MASTER from the initial environment. If UPLOAD, the objects will be moved from the FIRST environment, without being downloaded from the Master, and end at the MASTER. If set to U, the ISR will always UPLOAD. If set to N, the ISR will always download first, if old objects are linked to the ISR and then UPLOAD. The user will not have a choice if set to N or U.

3.4 Scan in Source Code – CC300 (CC500)

- All objects to be used with the Change Control need to be scanned in using option CC300.
- This creates an inventory list of the library name and the object name. This library name may be amended in each subsequent environment, using the Profiles ISR Library in CC002 or Individual Object Library in CC050.
- The environment number that the objects must be scanned into needs to be entered. However, this environment needs to coincide with the environment that the EspControl software is loaded into. This is normally the Development Environment. More than one Development environment



may be set up in CC001 as long as all the Development environments are on the local server. In this case, each environment's code must be scanned in and the number will differ according to CC001.

- i) One library must be scanned in at a time. Capture the library name. This is validated and if it does not exist, the user must check that the fuser path entered into CC001, for the selected environment, is correct.
- ii) To check that the directories are correct for scanning, change directory in Unix to the fuser path in CC001 and add on the library name and SRC to create the directory. View the list, and if correct, these will be scanned in properly. For e.g. .if the library being scanned in is ESPSOFT and the fuser path specified in CC001 is /usr/sag/nat/fuser_dev then
cd usr/sag/nat/fuser_dev/ESPSOFT/SRC to check objects in this directory.
- iii) If the Change Control software is not running in Development, but in for e.g. Production and the source code must still be scanned in from Development, then the Development server must be remote mounted onto Production to get the link correct for the scanning in of code in Development.
- iv) Source code need only be scanned once (at install time), as all new objects must then be scanned into option **CC050 (CC501)** using PF6.
- v) If new objects are scanned in via CC300, they will be given an "old-object" indicator and the migrations will not perform correctly. To reset this indicator to new, re-scan via PF6 in CC050 and set Re-New Obj to Y.
- vi) If the site already has code movement in Development and no code currently exists, except those objects being worked on, then this code must be restored to Development for the scanning in, and then the code can once again be removed.
- vii) To check that the inventory list is correct and before trying to migrate, go and check the object status list in **CC097 (CC503)**. Just run the option and the entire inventory list will be displayed. Remember, that this list will start with the INITIAL SCAN LIBRARY set up in CC001. To check other libraries, enter in Restart Library.

3.5 Initial Backup Routine – **CC350 (CC500)**

- If required, a new site must take an initial backup, or one doing an upgrade may prefer to take an initial backup of all objects in each environment.
- Due to the fact that the backup routine cycle (explained in detail in CC100 in the EspControl User Manual) backups objects in the FROM environment, the initial migration of each object will be done without a previous backup as the initial migration will be the first ISR linked to that object. In this scenario if the ISR abends, there will be no previous backup to restore to. Therefore, an initial backup routine should be executed, especially at the first install of EspControl.
- Note, that the above is not true for the MASTER (Production) environment, as the first time an object is migrated, it is first backed up in the Master before being downloaded. There will therefore be a previous backup, when the object is returned to the Master in the migration routine



and an error occurs. This is the case if the user is always going to use the DOWNLOAD option. If not, refer to point above.

- However, even in the above scenario the user has a choice to run the initial backup routine in Production as well, and for uniformity if an initial backup is run, it should be run in all environments.
- Note should be taken that this backup routine may take a while to run and should therefore be scheduled accordingly.
- The backup routine has many steps and rules, refer to the EspControl User Manual for the exact steps.
 - i) Remember to ensure that your backup paths exist, as if they do not, they will first be created before any backup is done, and this adds to the time.
 - ii) If the CODE MOVEMENT option is to be used, then ensure that all code exists, before running the initial backup and then this code may be scratched.
 - iii) Ensure that ALL necessary code has been scanned in. If not, then the backup routine will not back up the necessary objects as it uses the scanned code as its base as to what to back up.
 - iv) Select all the necessary environments to be backed up in the BACK* profile.
 - v) If for whatever reason, an object could not be backed up, the backup routine will continue and a report will be sent to the spooler at the end. Fix the problem and take a manual backup if necessary.
 - vi) After each environment has completed, go to CC200 and check the backup by entering the ISR number and select the option to DISPLAY the code of sample objects.
 - vii) The initial backup routine will ignore any NEW objects that have been scanned into the Inventory List.

3.6 Create and approve an ISR – CC030 (CC501)

- Once all the control variables and environments have been set up and the source code scanned in and an initial backup run if required, the next step would be to create and approve the ISR.
- The ISR must first be created with the linked profile, and then it must be ACCEPTED via CC040 (CC501). If the ISR is not accepted, then no objects may be linked to it and it cannot be migrated.
- The ISR may or may not have an XREF Number captured against it. If the code XREFNO is set to Y (CC010), then it will force the user to capture this number. This number is used to link the user's own project control system to the ISR in EspControl. Should any queries be needed against this number, there is a XREF-No enquiry CC093 (CC503) that will display all linked ISR's to a particular XREF-No and then the ISR History routine CC095 (CC503) may be called in return. A user exit ISNUX030 may be amended to validate this XREF-No.
 - i) Create the ISR and add all the required data fields
 - ii) Link the required PROFILE (ensure that this profile has all the environments and parameters necessary for the migration of the objects linked to this ISR)



- iii) Set the SECURITY indicator, if set to N, all ISR options will operate as normal. If set to Y, then only the originator of the ISR, may approve CC040, link **CC050 (CC501)** and upload approve via **CC060 (CC502)**. If user gets an error when approving, linking or migrating, check the security indicator on that specific ISR.
- iv) Get the applicable user to Accept or Reject the ISR via CC040.
- v) In CC040, link the applicable ISR Type to the ISR. This type is very important, although it is only used for information about the ISR; the type is also used as security with specific functions. The ISR Type may be viewed in the ISR CODES option **CC010 (CC500)**. Code UBK may only be used with ISR's linked to the Initial Backup **CC350 (CC500)**. Code OLD may only be used with Archive ISR's **CC250 (CC501)**. All other ISR types may be used with CC100.
- vi) If the Upload Approval has been marked as a parameter in the linked Profile, then the ISR will not migrate via CC100 until this ISR has been approved, for whichever environment is being migrated to or from. Use option **CC060 (CC502)** and enter the ISR number and Approve with a Y or Reject with a N.

3.7 Link Objects to an ISR – **CC050 (CC501)**

- Once an ISR has been created and approved, the objects to be linked to this ISR must be added via CC050.
- The ISR number entered in CC050 may not be linked to OLD or UBK ISR's.
- Every time the ISR number is entered, all the objects linked in the Inventory List will be displayed. The objects not linked to the entered ISR's, but to another ISR will be output only and may not be worked on. This ISR number linked to these objects will also be displayed. All the currently linked objects to the entered ISR will be displayed, with the object status and these objects may be amended if necessary. Any open objects may also be selected.
- The library for each environment will be displayed with an L in the LIB selection column. These libraries may be amended if necessary.
- The linked object will have an X in the selection column. If the user does not require this object to be linked to the ISR anymore, this X may be removed. The user must then decide to LEAVE the object in the environment where it has been moved to, or RESTORE the object through all environments back to Development. In this manner, an object may be de-selected without interfering with the code.
- The ISR may be copied to another ISR using **CC054 (CC501)**. To use CC054, the ISR being copied to, must first be created and approved. Then use CC054 and enter the ISR to copy From and the newly created one to be copied To. This option will link all objects that were linked in the From ISR to the new To ISR. It will not do any migration and will only copy the object if it is free and not linked to any other ISR. The history of all objects linked to the From ISR CC095 is used to link the objects to the To ISR. Note, these objects may be de-selected and new objects added after the copy, via CC050.



- i) Enter the ISR number
- ii) Link all objects using an X or deselect by removing the X
- iii) Check library per environment using an L – remember the rule – The library per environment is created as follows :
 - a) If the environment is TRUE Development, then the library may not be amended and is set up using the scanned in Library
 - b) The other environments, use the ISR LIBRARY set up in CC002, but if this is not to be used, update the actual object library in CC050 using selection L. The object library updated in CC050 will be used instead of the ISR Library.
 - c) If NO ISR Library or Object Library exists, then the scanned in library will be used.
- iv) If a new object is to be added, use PF6 and add the new objects, with the appropriate Library name. This object will be immediately added to the Inventory List and can be selected if required for the current ISR or may be selected in another ISR. This new object must reside in the environment selected for scanning into (for e.g. environment 1 in CC001), the same rule applies as to when scanning in the source code via CC300. The selected environment must be in the same fuser as to where the Change Control software is installed or have a pointer to it.
- v) If a copy of the object needs to be added to the inventory list and in this case to another Library – use PF1. This will copy the from object source to another library and add it to the current inventory. Only the source code will be copied and the user must stow the object manually.
- vi) During a call-out or emergency situation, where a developer needs to work on a Production module but does not have access to Production – use PF2 to add this module to another library. PF2 will not add the source as well, only the entry into the Inventory List and an ISR may be created to move it from the Production library to the Emergency Library to be worked on and then back again. This function does not scan the code but checks that the object being entered already exists in the Inventory List against another Library. These Emergency Libraries must be added against ISR Code STORECC in CC010 to be valid.
- vii) PF5 – similar to PF2 where the library entered is also checked against STORECC, but the object is actually scanned in via one of the Development Libraries, but is stored on the Inventory List against another Input Library, entered via PF5.
- viii) System error messages may also be migrated together with objects in an ISR. These are user error messages and they may be added to the ISR being used in CC050 by pressing PF4. Note, ensure the library for transfer is correct, same rules apply as the object library. Use L to check or amend. A system error may not be migrated in isolation in an ISR, there must also be objects linked. This can be overcome, by creating a “dummy” object in each environment for e.g. PGMERR and then this object may be used to migrate system error messages back and forth without linked any other objects.



3.8 Transfer an ISR – **CC100 (CC501)**

- Your ISR is now ready for transferring and must be done so using CC100.
- CC100 has many rules and regulations to adhere to, and uses the parameters set up in the Profile linked to the ISR being migrated, to do the transferring.
- CC100 must be executed ONCE for each environment linked to the profile. It will follow the path of the profile and copy, backup and restore according to the parameters. See EspControl User Manual for in depth detail to all these parameters. The history routine CC095 will display the migration path, as well as what parameters were used for the user to check on.
- If the PATH CHOICE is N in the linked profile, the ISR will download from the Profile Master to the Profile Initial environment and then back up the line to be completed once ALL the objects have been retransferred to the Profile Master.
- If the PATH CHOICE is Y, then the user may decide whether to DOWNLOAD or UPLOAD. If DOWNLOAD follow same rules as above. If UPLOAD then the objects will start the migration path from the Profile Initial environment and then complete once ALL objects have been transferred to the Profile Master.
- If the PATH CHOICE is U, then the ISR will upload from the Initial environment automatically skipping the download move.
- If System Error messages have been linked OR the Source Unload indicator marked in the Profile, then the transferring of the system error message and/or the unloading of the source to a text file will occur at the end of the current environment move. These functions have reports sent to the spooler if any error occurs. If an error occurs, the user must correct then rerun CC100 and the migration will ignore ALL objects and only do the transferring of either the system error message or source unload depending on what indicators are set.
- If the system error message or source unload in error, is no longer required, and you want to bypass this error, then remove the ISR from “error” status by using **CC076 (CC502)** for source unload errors and **CC077 (CC502)** for system error messages. This will then ignore the transfer of the error message and/or source unload when re-running CC100.
- The migration of the objects will be displayed on the screen in CC100 and may be viewed as the transferring of objects take place. ‘
- If an ISR abends, the error codes and details may be viewed in CC095, as well as in the log files specified at the start of this guide. Many scripts are used in CC100 and the full error of these may be viewed in the log files in cd \$ESPCCTMP.
- The object status will equal the environment number (by using the same number as selected for the path on the profile) that the object is in. 99 means in the Master. As each profile has a different path, object status 02 could be a different environment to status 02 in another ISR. To check this out, use option **CC097 (CC503)** to see the object status and then check the PROFILE in CC002 to match this up. Alternatively check **CC090 (CC503)** which will reflect all objects linked to the entered ISR and give a description of the status.



- At any given time an ISR may have objects of differing statuses. This may be because an object was linked after a migration has taken place, or an object was returned to a previous environment to be worked on, before again being transferred up. CC100 will ALWAYS move the objects in the lowest object status or if the newly added object is in the Master environment first, before continuing with the next migration. If CC100 is executed whereby lower status objects are moved, all the other objects are ignored during this migration. The rule being – The next environment WILL NOT be transferred to, until all objects linked in the ISR have the same object status.
- The ISR status will reflect the object status in the lowest form.
- The very first function in a migration is the PATH VALIDATION. This PATH VALIDATION forms a summary file of all linked objects and their libraries, using the rule as specified above. The routine then validates the fuser and backup paths and the existence of objects BEFORE actually copying or moving source. If this validation fails, the ISR will HALT and the user must check CC095 to see what the validation errors or objects in error were. An Error Object report may also be printed if required. CC100 cannot continue until the problems have been solved and the ISR has been removed from HOLD status. To do this, use **CC075 (CC502)** and enter ISR number.
- If the ISR has a linked profile using CODE MOVEMENT, i.e. ALLOW-SOURCE = M, then the object is scratched after it has been migrated successfully to the next environment. In this situation there is an OBJECT EXISTANCE test for each object in the ISR. If the object exists, the user will get a warning, as it should not be in the environment and therefore some problem could exist. The user may IGNORE and OVERWRITE, REJECT that particular object from the current ISR movement, or HALT the entire ISR to investigate.
- If the user requires the object to be RETURNed to the previous environment, not RESTORED but returned in its AMENDED state, then the source may be returned by using the RETURN ISR function **CC102 (CC504)**. This function will return the actual source to the previous environment for further amendments before moving up again. See rules in EspControl User Manual for this function. Once the objects have been linked, they may only run via CC102 once the Return ISR has been approved for Return, use option **CC061 (CC504)**.
- If the user does not require a move to a specific environment, with the exception of the Master and Initial, then the skip function **CC105 (CC501)** may be used. This function will skip an environment completely and transfer to the next-plus-one environment, without the user having to either complete the ISR or unlink the objects and choose another profile linked to a new ISR to do the move. See rules in EspControl User Manual for this function.
- Depending on the RESTORE INDICATOR for the transferred ISR, if an error occurs and this is set to Y the ISR will halt, display the error code and restore all objects back to their previous object status and the last backup taken in that particular ISR. If no backup for that ISR exists, the previous backup for any ISR that was last linked to the object will be used. The history enquiry will reflect all details necessary. If the indicator is set to N, no halt or restore will take place. The



ISR will complete its current migration and a report of all errors sent to the spooler. These are also available in option CC095.

- If a specific object is being queried and the history of ALL ISR's linked to this object is needed, use **CC088 (CC503)** to view all related ISR numbers and the related history. NOTE – this function is only available from version 6.10, so in an upgrade situation detail for this option will only be valid for objects migrated in V6.10 (from upgrade date and time).
- If the Audit Report option is set to Y (ISR code AUDITRPT in CC010), then after each migration move an ISR Audit Report will be sent to the spooler detailing all the objects moved and the relevant indicators or errors (as viewed on the screen in CC100).

3.9 Complete an ISR – **CC070 (CC502)**

- When an ISR has been through the entire migration cycle created by the linked Profile, the completion routine CC070 is automatically called.
- This routine may be called manually as well, if the user wishes to complete an ISR before the end of its cycle. If the routine is run before the actual completion, the user will get the choice to RESTORE objects or leave them in their current environments AS IS. Note – NO RESTORE must be Y for the restore of the objects to be ignored. The user will be given a warning if marked as N, use PF3 to mark with Y if necessary, or enter for restore to begin. If marked with N, all objects will be restored to the initial environment using the backup of the completed ISR. If marked with Y, all objects will remain in their current environment as they were transferred there.
- In both the options above, all objects will be unlinked from the ISR for further use, the ISR will be completed and get a status of 99 and all the object statuses will equal 0.
 - i) Enter ISR number, if already completed error message will occur
 - ii) Enter Y or N for NO RESTORE
 - iii) Enter number of hours project took for total migration cycle.
- Use option **CC080 (CC503)** to view ISR status and **CC090 (CC503)** to view objects linked to the ISR if any queries arise. Obviously, the entire ISR event will have been recorded in CC095 history.
- **The migration cycle has now been completed and no further action is required, except to start again with CC030.**



4. Extra Routines

4.1 Archiving – CC250 (CC501)

- Once a site has no further use of an object, this object must not be scratched manually, but archived using CC250.
- This routine will scratch the object inked to the ISR, remove the object from the Inventory List and backup the object using the Archive ISR number. If scratched manually, the link in CC050 will remain and the user will still be able to select this object even though it does not exist.
- As an archived object will now be missing from all object enquiries, if any queries arise for a particular archived object, use the object history enquiry **CC088 (CC503)** and if the object is not reflected on the list, press PF6 and enter the archived object name. CC088 will in turn display all previously linked ISR's via CC090 and in turn all history via CC095 using the reflected ISR numbers.
- An object may be archived in any environment and these environments will be displayed depending on the profile linked to the Archive ISR. The user may select one or more of these environments and the object will be scratched and backed up in all selected environments. NOTE – if the object does not exist in a specific environment, the Archiving will continue as normal and just remove the object from the Inventory List.
 - i) Create an ISR via CC030 and link a profile with the correct environments
 - ii) Accept via CC040 and ensure that ISR type OLD has been selected, else CC250 will give an error
 - iii) Run CC250 and select environments from the window displayed, then select required objects from the Inventory List
 - iv) Press PF4 for the archiving to begin
- View History CC095 to see details of archived routine by using the archive ISR No and use **CC200 (CC504)** to actually view the backup source if required to do so.

4.2 Restores – CC200 (CC504)

- There are three main restore functions, but the most powerful restore function is CC200. This function allows the user to restore to any environment, any library and any backed up code, within the number of versions allowed. (See CC001 for Max Versions)
- The restore functions **CC110 (CC504)** and **CC115 (CC504)** will only restore source code in the current ISR being specified in these two functions. Once an ISR is complete these two functions are out of scope and CC200 must be used.
- To fully understand the restore routines, the user needs to fully understand the backup cycle. This backup cycle is explained in detail in the EspControl User Manual and should be read before attempting the restore functions.



- The main rule to the backup cycle is that the backup is taken when objects are being migrated FROM environments and NOT before the code is overwritten. This means that a backup in an ISR will refer to the source that was amended in the ISR and not the source before the migration.
- Therefore, the restore during a migration cycle, when an error occurs or functions CC110 and CC115 will actually restore the code that was created in the ISR being worked on. The reason for this is the CODE MOVEMENT option will scratch code and so the restore will return the code that was created in the ISR to be worked on. Basically, if code movement is not being used, then the only function of the above-mentioned restore routines is to return the ISR and object status to the previous status and the source to the last source that was in the environment before the move, and therefore CC100 may again be run, re-migrating the source code. In other words, the move From will restore the actual source being moved, and the move TO will restore the source that was there before the move From.
- If the user does not require the above restores, then CC200 should be used instead, whereby the ISR to be restored to, may be selected by the user. The user may then select the environment and the library to restore to.
- If the user requires different restores to CC110 or CC115, but also requires the object status to be restored, then CC110 or CC115 must be run to restore the object status in conjunction with CC200 to get the correct version of source required. Unlinking the objects via CC050 will also cause the status to be returned and this can be done with No Restore.
- Note, using CC200 gives the user a chance to view the source before restoring it, to check that the correct version will be restored, use option D for Display
- For CC200
 - i) Select the objects to be restored
 - ii) Select one or more environments
 - iii) Select D for Display (this will be reflected on the screen) or R for Restore
 - iv) If D selected PF3 to exit when completed viewing. Enter "?" in the command line to view a list of commands for the viewing of the source.
 - v) If R, confirm the restore and select the library to be restored to. The default library is RESTORE, but this may be amended to any library that exists in the selected environment. It may be a good idea to use library RESTORE when restoring to Production, so as to not interfere with the users.
 - vi) Only the source code is restored and must be stowed by the user to create the object code again.



4.3 ISR Query Functions - **CC503**

- There are many ISR query functions available, many of which have been mentioned in this guide above. All of these functions and routines are available on menu CC504. See EspControl User Manual for further detail.
- Should the user require any additional information regarding the ISR or the actual Objects in the ISR or even objects outside of the ISR, this menu will enable the user to find out whatever he needs to.
- **CC088** – Object History and **CC095** – ISR History will reflect all the detail needed to know what happened in the past to an object and what happened in the migration routine to an ISR
- **CC093** – X-Ref History will be of importance if the user has linked his own Project Control number to the Change Control system.
- **CC095** – ISR History function reflecting detail as specified above, as well as an Audit function by pressing PF5. This will produce either a report or a .csv file detailing all audit moves in the time period and environments specified by the user.