Using CA IDMS™ System Exits

Mike Picchioni & Jake Hart
CA Technologies

CA IDMS™ Technical Conference

Framingham MA
December 2-5, 2014

Abstract

This session describes the system exits available in CA IDMS™. Practical uses for the exits are discussed as well as coding conventions.

Agenda

- Overview
- Named Exits
- Numbered Exits
- Changes in Exits
- Sample Exits

Overview

- Why use an Exit ?
  - Allow for CA IDMS site specific customization
  - Predefined exit points allow for control at critical processing points
- CA IDMS DC/UCF Named exits
  - Allow users programs to receive control during system operations
- CA IDMS/DB Named exits
  - Allow user programs to receive control during database operations
- CA IDMS Numbered exits
  - Allow users programs to receive control during system operations

CA IDMS DC/UCF Named Exits

Entry Point	DC/UCF Module	Exit Usage
IDMSSVCX	The CA IDMS SVC or equivalent module	Capture transaction accounting statistics
TCKREXIT	DC/UCF startup module	Given control when the TICKER task is woken up
USRIDXIT	IDMSSTRT, UCFCICS, or IDMSINTC	To modify the user who is to sign on to the DC/UCF system

CA IDMS DC/UCF Named Exits (cont.)

Entry Point	DC/UCF Module	Exit Usage
WAITEXIT	DC/UCF startup module	Monitor operating system waits
WTOEXIT	IDMSOS00 or the DC/UCF startup module	Review DC/UCF messages
WTOREXIT	IDMSOS00 or the DC/UCF startup module	Modify operator communications

CA IDMS DB Named Exits

Entry Point	CA IDMS/DB Module	Exit Usage
IDMSAJNX	Archive Journal	Review journal records
IDMSCLCX	IDMSDBMS	Compute a CALC key target page
IDMSDPLX	IDMSDBIO	Maintain duplicate journal and/or database files

CA IDMS DB Named Exits

Entry Point	CA IDMS/DB Module	Exit Usage
IDMSIOXT	IDMSDBIO	Disk read/write
IDMSIOX2	IDMSDBIO	I/O processing
IDMSJNL2	IDMSDBIO	Journal buffer is written to journal file

Installing Named Exits

- Named exits must have specific Entry Point names
- Named exits are linked with IDMSUXIT as of release 18.0
- IDMSUXIT must be linked with all enabled named exits
  - Entry point for IDMSUXIT is UEXITS
- Sample JCL is in the CAGJSAMP install library
  - PTF RO75785 for 18.0
  - PTF RO71145 for 18.5

CA IDMS Numbered Exits

- Numbered exits are user exits that you define via module RHDCUXIT
- RHDCUXIT is loaded at system startup as part of the nucleus, and part of the batch and non-TP monitor interface
- Numbered exits are divided into two types
  - System-invoked exits
  - User-invoked exits

CA IDMS Numbered Exits (cont.)

System-invoked exits

(0-255)

- System-invoked exits provide DC/UCF functions such as signon, signoff, and security checking
- DC/UCF determines the routine that calls the exit. These exits are invoked at predefined logical point in DC/UCF modules.

CA IDMS Numbered Exits (cont.)

User-invoked exits

(256-above)

- User-invoked exits are invoked by site-written programs
- For example, a user written assembler program may call a user written exit 256 by means of the #XIT macro

CA IDMS Numbered Exits (cont.)

Exit	Function
Exit 0	System initialization
Exit 1	Signon
Exit 2	Signoff
Exit 4	New task
Exit 5	Task termination I (before statistics are written)
Exit 6	Task termination II (after statistics are written)
Exit 7	Write-to-log

CA IDMS Numbered Exits (cont.)

Exit	Function
Exit 8	Log full
Exit 9	System statistics
Exit 12	Terminal I/O error
Exit 13	Shutdown
Exit 14	Bind Run Unit and Ready Area
Exit 15	VIB statistics
Exit 16	Write Printer

CA IDMS Numbered Exits (cont.)

Exit	Function
Exit 17	Input data stream
Exit 18	Output data stream
Exit 19	Asynchronous terminal connection
Exit 20	Resource limit
Exit 21	SYSOUTL report
Exit 22	Report security and routing
Exit 23	Pre-BIND Run Unit

CA IDMS Numbered Exits (cont.)

Exit	Function
Exit 24	GET TIME
Exit 26	OLQ JCL
Exit 27	ERE extension examiner
Exit 28	Security preprocessing
Exit 29	Security postprocessing
Exit 30	Victim selection for deadlock detection
Exit 31	Transaction statistics

CA IDMS Numbered Exits (cont.)

Exit	Function
Exit 32	SYSOUTL detail record
Exit 33	Program loader
Exit 34	Unqualified DBKEY Find/Obtain
Exit 35	Stalled task information
Exit 36	Global deadlock victim selection
Exit 37	Recovery wait
Exit 38	Quiesce area

CA IDMS Numbered Exits (cont.)

Exit	Function
Exit 39	SQL Syntax Collection

Calling Conventions

- DC/UCF Calling Conventions
  - When using these conventions, you must use the #START macro to begin the exit routine and the #RTN macro to end the exit routine
  - On entry to the exit routine the following registers contain information:

R13	Next available TCE stack entry. Accessible via macro #GETSTK
R12	Base address for module
R10	Address of the common system area (CSA). The value in this register must not be modified.
R9	Address of the task control element (TCE). The value in this register must not be modified

Calling Conventions (cont.)

- General Calling Conventions
  - Apply under DC/UCF and IBM calling conventions
  - These calling conventions apply to all numbered exits
  - On entry to the numbered exit, register 1 points to a two-word parameter list:

Word One	The address of a fullword containing the exit number
Word Two	Either the address of an exit specific parameter list or 0 (zero) if no parameters are passed

- On return, R15 should contain the return code

Required Macros for DC/UCF

- #MOPT
  - Sets up options for the issuing module
  - The only code to appear prior to #MOPT is 'TITLE' comments, or source macro definitions
- #START
  - The first instruction in a user-exit routine that uses DC/UCF calling conventions
  - This macro is additionally used to specify the MPMODE to be assigned to the exit routine
  - Modules which use DC/UCF calling conventions should be assigned an MPMODE consistent with the control blocks accessed

Required Macros for DC/UCF (cont.)

- #RTN
  - Used to terminate a routine and returns control to the calling routine
  - #RTN must be the last instruction executed in a user-exit routine that uses DC/UCF calling conventions
- #GETSTK
  - Used in a system mode exit to acquire storage from the TCE stack area; this can be useful in preserving reentrancy

Installing numbered exits

- Code the user-exit routine
- Install the numbered exit by including an entry for the #DEFXIT macro in module RHDCUXIT
  - Can use Entry Point of program or Name of program
- If #DEFXIT coded with Entry Point
  - Assemble and Link module RHDCUXIT
- If #DEFXIT coded with name
  - Define the exit routine program to the system SYSGEN

Assemble and Link edit RHDCUXIT

- Assembler and link edit module RHDCUXIT to make it available for nucleus loading at startup
- Include any routines which are specified by #DEFXIT macros with an Entry Point
- Specify that the RHDCUXIT entry point is UXITEP1
- Sample JCL is in the CAGJSAMP install library

Coding the #DEFXIT Macro

- The #DEFXIT macro declares exit information to DC/UCF, this includes the exit's name, entry point, mode, addressing etc.
- Place #DEFXIT macros in numerical order in the UXIT module, according to the number of each associated exit
- Include placeholder #DEFXIT macros for skipped numbered exits
  - The placeholders will allow the #DEFXIT source to accurately reflect the position of each numbered exit

Coding the #DEFXIT Macro (cont.)

```
.
.
#DEFXIT                                                    exit 00
#DEFXIT   MODE=SYSTEM,CALL=DC,EP=XSON,AMODE=ANY            exit 01
#DEFXIT                                                    exit 02
#DEFXIT                                                    exit 03
#DEFXIT                                                    exit 04
#DEFXIT   MODE=SYSTEM,CALL=DC,EP=XTASK,AMODE=ANY           exit 05
```

Changes in Exits

- Tools Exits
- CICS Exits
- Hooked Exits

Tools Exits

- Tools Exits are no longer linked with RHDCUXIT (as of R18.0)
  - CA ADS™ Alive
    - No longer uses exit 333
  - Task Analyzer and Master Key
    - Require SYSIDMS parameters in startup JCL to enable product features
    - TASK_ANALYZER_EXITS=ON MASTERKEY_EXITS=ON
  - CA IDMS™ DML Online (DMLO)
    - If USEREXIT=YES in customization macro USDTPARM then exit USDMLXIT is invoked
    - Sample code is in CUSTOM.SRCLIB

CICS Exits

- No longer linked with the interface
- Now linked with IDMSINTC or IDMSINTL
- When linking IDMSINTC or IDMSINTL, must INCLUDE your exits

CICS Exits

Entry Point	CA IDMS/DB Module	Exit Usage
IDMSCEON	IDMSINTC	To gain control upon entry into IDMSINTC interface before the requested function is passed along to the appropriate client.
IDMSCEOX	IDMSINTC	To gain control upon return from processing the requested function and before control is returned to the caller of IDMSINTC.
OPTIQXIT	IDMSINTC	To alter dynamically so that an individual SQL session can be routed to a specific back-end CV
OPTIXIT	IDMSINTC, IDMSINTL	To alter dynamically the location where the request will be processed

Hooked Exits

- New functionality for early 2015
- Allows user programs to gain control before or after a CA IDMS vector gets control
- Can prevent a CA IDMS vector from gaining control
- Implemented through a #HOOK macro call

Sample DC/UCF Exits

- Review of user coded Numbered Exits
- Additional Handout

Summary

- Overview
- Named Exits
- Numbered Exits
- Changes in Exits
- Sample Exits

FOR INFORMATION PURPOSES ONLY

Terms of this Presentation

This presentation was based on current information and resource allocations as of December 2014 and is subject to change or withdrawal by CA at any time without notice. Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

Certain information in this presentation may outline CA's general product direction. All information in this presentation is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this presentation "as is" without warranty of any kind, including without limitation, any implied warranties or merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised in advance of the possibility of such damages. CA confidential and proprietary. No unauthorized copying or distribution permitted.

Questions and Answers