

# Special Natural CICS Functionality

This chapter of the Natural CICS Interface documentation explains special Natural CICS functionality.

It covers the following sections:

- Calling Non-Natural Programs
  - Dummy Screen I/Os with Natural under CICS
  - NCISTART - Natural CICS Nucleus
- 

## Calling Non-Natural Programs

One of the first actions a Natural task does at its start, is to activate an exit for abnormal termination processing. This exit is used to release all resources including the thread in the case of an abnormal termination. Therefore, a non-Natural program must not issue EXEC CICS ABEND CANCEL or the equivalent macro level request, as such a request cancels the current session ignoring any active exit. If so, Natural is not able to clean up its resources, and the thread and the roll facility are not released.

A thread is assigned to a Natural task whenever a Natural program is active. This is also true when non-Natural programs are called (following CICS linkage conventions).

Therefore, such programs should not do excessive I/Os and other work load without Natural receiving control in between. If a non-Natural program is doing conversational screen I/Os, you can code a SET CONTROL 'P=V' statement in the Natural program that calls the non-Natural program before the calling statement: this indicates that parameter data are copied out of the thread and the session is rolled out before calling the non-Natural program.

## Calling Non-Natural Programs via Standard Linkage Conventions

A non-Natural program is invoked (CALLED) by Natural in the way programs are invoked within the underlying operating and/or TP-monitor system.

In CICS, non-Natural programs are invoked by means of EXEC CICS LINK requests. However, when, for example, the same subroutine program (not issuing any CICS or operating system request) is to be used for both batch and online processing, a non-Natural program may also be invoked by using CICS standard linkage conventions; that is, via BALR R14 ,R15.

For further information, see the terminal command %P=S in the *Terminal Commands documentation*. See also the parameter SLCALL in macro NCMPRM.

## Calling Non-Natural Programs with Parameter Values in a COMMAREA

By default, non-Natural programs are called with the addresses of the request parameter and field descriptor lists (R1 and R2; see also the description of the CALL statement in the *Natural Statements documentation*) passed in the TWA and COMMAREA (depending on the setting of the NCIPARM parameter COMACAL).

A more CICS-like method is to pass the parameter values in a CICS COMMAREA, particularly when the called program resides in another CICS region - Distributed Program Link (DPL) -, as addresses within the "calling" region are not accessible by the "called" region.

For details and restrictions, see the terminal command %P=C in the *Terminal Commands* documentation.

### Prerequisite:

This functionality requires COMACAL to be set to YES in NCIPARM.

When the parameter values are passed to a CICS COMMAREA, the first two words of the CICS TWA are low-value, which means that no R1 and R2 information is passed.

## Dummy Screen I/Os with Natural under CICS

If a SET CONTROL 'QO' statement is placed before a Natural statement that causes a screen I/O, the terminal output is not executed by Natural under CICS. Consequently, both the Enter key and user input are not passed back to Natural.

This functionality may be useful in the following situations:

1. When leaving pseudo-conversational screen I/Os to non-Natural programs called by Natural. The non-Natural program performs the EXEC CICS SEND operation and returns to Natural. Due to the SET CONTROL 'QO' statement, the next Natural screen I/O terminates the task of a pseudo-conversational session. Upon screen input, Natural receives control and invokes the non-Natural program again, which then performs the EXEC CICS RECEIVE.
2. When changing the Natural pseudo-conversational transaction ID "in-flight" without requiring a terminal operator intervention:

```
MOVE *INIT-ID TO termid
CALLNAT 'CMTRNSET' trnid /* change the restart transaction ID

* starting a task on your terminal forces an interrupt as if
* pressing any attention identifier

CALL 'CMTASK' USING trnid
H'0000' H'00000000' termid
SET CONTROL 'QO'
INPUT 'DUMMY' /* dummy I/O, which you will never see
WRITE 'HELLO' *INIT-PROGRAM /* now the new transaction ID is active
```

3. When switching to an application outside Natural, perhaps even in another CICS AOR (application-owning region), without requiring a terminal operator intervention:

```
* starting a task on your terminal forces an interrupt as if
* pressing any attention identifier

CALL 'CMTASK' USING trnid data-length start-data termid
SET CONTROL 'QO'
INPUT 'DUMMY' /* dummy I/O, which you will never see
WRITE 'HELLO' *INIT-PROGRAM /* now the new transaction ID is active
```

In this case, it is the responsibility of the application being invoked to stack the Natural restart data when they are passed in a CICS COMMAREA, as a COMMAREA most likely is used by the new (pseudo-conversational) application, too.

## NCISTART - Natural CICS Nucleus

NCISTART (that is, the Natural CICS nucleus with NCISTART as entry point) is eligible to be placed into the CICS PLTSD for CICS quiesce stage 1 or 2 execution.

- When executed in quiesce stage 1, NCISTART performs the SYSTP snapshot function (as described in *SYSTP Utility* in the Natural *Utilities* documentation).
- When executed in quiesce stage 2, NCISTART force-terminates all active Natural sessions prior to performing the SYSTP snapshot function.

NCISTART holds logic to be called (via a CICS LINK) by a node error program with the relevant CICS terminal entry address either in the CICS COMMAREA (with CICS/TS) or at the beginning of the TWA (with CICS/VSE Version 2.3).