Invoking Natural from User Programs

This section of the Natural CICS Interface documentation describes the various ways of how Natural can be invoked from user programs.

It covers the following topics:

- Commands for Activating a Natural Session

- Front-End Parameters

- Front-End Invoked via LINK

- Front-End Invoked via RETURN IMMEDIATE

- Front-End Invoked via START

- Front-End Invoked via XCTL

- Front-End Invoked via Distributed Program Link (DPL)

- Invoking Front-End Program as Back-End

Commands for Activating a Natural Session

This section covers the following topics

- Using EXEC CICS XCTL or EXEC CICS LINK

- Using EXEC CICS RETURN IMMEDIATE

- Using EXEC CICS START

- Using Distributed Program Link (DPL)

- Sample Programs

- Using the External Subroutine CMTASK

A Natural session can be activated by user front-end programs with one of the following commands:

- `EXEC CICS XCTL`

- `EXEC CICS LINK`

- `EXEC CICS RETURN IMMEDIATE`

- `EXEC CICS START`

or the equivalent CICS macro level requests.

Using EXEC CICS XCTL or EXEC CICS LINK

When using `EXEC CICS XCTL/LINK`, the parameters used by Natural can be passed in a CICS COMMAREA or in the TWA.

- Natural determines the location of the startup parameters by inspecting the length of the COMMAREA passed to it during session initialization.

- If the length is 22, Natural tries to locate the parameters in the COMMAREA, otherwise it is assumed that they have been passed in the TWA.

To identify a front-end program properly, it is mandatory that the first 4 bytes of the front-end parameter list represent the current transaction ID.

The transaction ID associated with the front-end program must have a TWA size that is equal to or greater than the Natural TWA size; see also *ncitransact* (*Step 20* in the section *Installing the Natural CICS Interface* of the Natural *Installation* documentation).

Using EXEC CICS RETURN IMMEDIATE

When using `EXEC CICS RETURN IMMEDIATE`, the front-end parameters used by Natural can be passed in a CICS COMMAREA and the dynamic parameters used by Natural can be passed with `INPUTMSG (...)` and `INPUTMSGLEN (...)` of the `EXEC CICS RETURN IMMEDIATE` command.

Using EXEC CICS START

When using `EXEC CICS START`, the front-end and dynamic parameters used by Natural can be passed with `FROM (...)` and `LENGTH (...)` of the `EXEC CICS START` command. The parameters are described on the following page.

Using Distributed Program Link (DPL)

When using `EXEC CICS LINK` with the `SYSID` parameter pointing to a remote region, the front-end and dynamic parameters used by Natural have to be passed in a CICS COMMAREA. In addition also a `TRANSID` parameter has to be specified naming the transaction code of a mirror transaction with a TWA size satisfying Natural's requirements.

It should be noted that the same restrictions as for asynchronous Natural sessions apply to Natural sessions invoked via DPL, that is, no input is possible (therefor the same dynamic parameter settings are recommended) and the caller just gets control back after Natural session termination.

Sample Programs

A series of sample programs for the various programming techniques is supplied in the Natural CICS source library; see also *Natural CICS Sample Programs*

Using the External Subroutine CMTASK

It is possible to start a Natural session from a Natural program by calling the external subroutine CMTASK. Refer to the sample Natural program ASYNCICS in library SYSEXTP.

Front-End Parameters

The following list of parameters must be supplied to invoke Natural from a user front-end program:

Pos.	Contents
1 – 4	**Invoking transaction ID** This value must be equal to the current transaction ID. Via the invoking transaction ID, Natural identifies that it was called by a user front-end program. When being called with XCTL, the transaction is restarted at the end of the Natural session via RETURN with TRANS ID, unless a return program name is specified (see 5th parameter).
5 – 8	**Address/offset of dynamic parameter string** If dynamic parameter overwrites are to be evaluated, this value should be set to the address located 12 bytes before the dynamic parameter assignment string. When being called with START or DPL, the field must be set to the offset (relative to the start of the front-end parameter list) of the address located 12 bytes before the dynamic parameter assignment string.
9 – 10	**Length of the dynamic parameter string** Zero indicates that no parameters are to be passed. 32760 is the maximum length allowed. If the maximum value is exceeded, the session is terminated with a corresponding error message.
11 – 14	**Natural transaction ID** The value specified is the transaction ID to be used for controlling a pseudo-conversational Natural session, when being called with START or XCTL. This transaction is invoked each time the Natural session is restarted in pseudo-conversational mode; that is, with each terminal I/O. If the Natural transaction ID is not specified, Natural restarts the transaction ID which initiated the current CICS task, and the front-end program regains control after each pseudo-conversational I/O.
15 – 22	**Back-end program name** This 8-byte value is the program name to which control is transferred at the end of the Natural session with a CICS XCTL command, rather than restarting the calling transaction ID via RETURN with TRANSID. If this field is numeric in the first byte, Natural simply RETURNs without activating any back-end. Please note that this field can be superseded by the Natural profile parameter PROGRAM. For the conventions of calling non-Natural back-end programs, refer to the Natural Operations documentation.

Front-End Invoked via LINK

On return to the front-end, Natural indicates in the TWA if the session has terminated or not: when the session has terminated, the TWA holds regular back-end information (see Back-end Program Calling Conventions in the *Operations* documentation), else Natural puts the NEXTTRANSID into the first four bytes of the TWA.

If Natural is running in pseudo-conversational mode (profile parameter PSEUDO set to ON) and has been invoked by EXEC CICS LINK (or the equivalent CICS macro level request), the original invoking transaction is invoked each time Natural writes to a terminal and waits for input, which means that Natural issues a "logical" CICS RETURN TRANSID (..) after having written its restart information into CICS temporary storage.

The invoking transaction must recognize this situation (for example, by checking whether a NEXTTRANSID has been sent or by the existence of NCOM*xxxx* TS records - where NCOM is the Natural CICS parameter generation option and *xxxx* is the terminal ID -) and pass control back to Natural.

The advantage of this method is that, during the session, the front-end program can decide to pass control to another application (for example, COBOL) and to resume the Natural session later.

For further details see the PSEUDO parameter description in the *Parameter Reference* documentation.

Per design, Natural treats a LINK front-end program as a back-end program at session termination, that is, the *Back-End Program Calling Conventions* apply.

In CICSplex Environments

Make sure that the NCOM*xxxx* TS records can be accessed by all participating CICS AORs (for example via appropriate CICS TST definitions).

Alternatively the LINK front-end program can also pick up the NCI session restart information in CICS temporary storage on task termination and pass it in a CICS COMMAREA by itself; such a COMMAREA has then to be put into CICS temporary storage again prior to invoking Natural for session resume.

Front-End Invoked via RETURN IMMEDIATE

This front-end technique only works for terminal-bound Natural sessions. Natural scans for start-up parameters supplied with the COMMAREA. Note that when using this technique, potential dynamic parameters cannot be passed chained to the front-end parameters, that is, the dynamic parameters' address fields must be zero. Instead, potential dynamic parameters can be passed via terminal input data, which are obtained by Natural by an EXEC CICS RECEIVE command.

Front-End Invoked via START

If the Natural session is a started task (that is, invoked by an EXEC CICS START or EXEC CICS LINK/XCTL command by a front-end user program which has been STARTed), Natural first scans for startup parameters supplied with the COMMAREA, then it scans for parameters in the TWA and finally it tries to obtain the necessary parameters by an EXEC CICS RETRIEVE command.

Front-End Invoked via XCTL

If the Natural session is initiated from a front-end program with XCTL and no return program is specified (that is, neither a fifth parameter in the session startup parameters nor a PROGRAM specification in the Natural dynamic parameters or the NTPRM macro), Natural restarts the user front-end transaction at session termination via RETURN with TRANSID by internally simulating a PROGRAM='RET=*xxx*' specification, with *xxx* being the front-end transaction code.

To avoid a loop condition, logic must be included into the user front-end routine to decide whether a new session is to be started or an old session is to be resumed.

Front-End Invoked via Distributed Program Link (DPL)

If the Natural session is invoked via DPL, Natural first determines if it is directly invoked in the server region or indirectly via EXEC CICS LINK/XCTL by a local front-end program. When being invoked directly, Natural retrieves the start parameters from the CICS COMMAREA. When being invoked indirectly, Natural scans for startup parameters supplied with the COMMAREA, then it scans for parameters in the TWA. On return Natural passes regular back-end data in the TWA when there is a local LINK front-end program available, otherwise it returns the termination message and potential back-end data in the remote client's COMMAREA.

Invoking Front-End Program as Back-End

If the Natural session is initiated from a front-end program and this program is also specified to be the return program, the user front-end should also check for the initiating transaction ID.

In particular this applies if the front-end program is not in pseudo-conversational mode but Natural is in conversational mode.

In this case Natural is invoked again rather than getting terminated, but this time without detecting that it is called by a front-end program, as the first parameter in the startup parameters is the Natural transaction ID.