Natural under Com-plete

This section describes how to operate Natural in a Com-plete environment.

It covers the following topics:

- Use of the Abend Exits
- Storage Usage
- Support of Back-end Programs
- Com-plete Support in Natural Batch Runs
- Support of Asynchronous Natural Processing
- Invoking Natural from User Programs
- Storage Thread Key Handling
- Support of User Exit Handling during Session Initialization

See also:

- For further details of the Com-plete product, refer to the Com-plete documentation set.
- For details concerning the following topics, refer to the Natural Installation Guide for Mainframes:
  - Structure and Functionality of the Natural Com-plete Interface
  - Prerequisites
  - Installation Tape for the Natural Complete Interface
  - Installation Procedure for the Natural Complete Interface
  - Using a Natural Local Buffer Pool under Com-plete
  - Using the Com-plete *ULIB Function
  - Installation Verification
  - Customizing a Natural Com-plete Environment
- The Natural utility SYSTP provides various TP-monitor-specific functions (see SYSTP Utility).
- See also Natural under Com-plete Abend Codes and Error Messages.

Use of the Abend Exits

The SPIE and ABEXIT exits can generally be deactivated by setting SPIEA=NO in NCFPARM.
The ABEXIT exit is called during Com-plete's EOJ handling for an abnormal program termination other than a 0CX abend to clean up processing.

By default, an 0CX abend is interpreted by the SPIE exit routine.

- Running with DU=ON, the Natural session is dumped and correctly terminated with error message NAT9974.
- Running with DU=FORCE, the SPIE and ABEXIT exit routines are disabled, an immediate dump during Com-plete is produced.

If DU=OFF, Natural responds with error message NAT0954, NAT0955 or NAT0956, and the entire abend PSW and registers 0 to 15 are contained in the IOCB at offset x'290'.

Note: DU=SNAP is currently not supported for COM51, but only with the next Com-plete version.

Storage Usage

At session initialization, the amount of space defined with parameter NTHSIZE in NCFPARM is allocated as thread GETMAIN above or below the 16 MB line, depending on the parameter THABOVE, for usage by Natural.

The WPSIZE profile parameter determines the sizes of below and above work pools. By default, the size of the below subpool is set to 32 KB.

Therefore, you must catalog the Natural Com-plete front part with the Com-plete utility ULIB, RG size = 36KB or larger.

The remaining areas within the Com-plete thread parts below and/or above (Com-plete ULIB RG= specification and/or THABOVESIZE= specification) are used by Com-plete for the following things:

- user subroutines;
- the extended program interrupt element (EPIE) in OS/390;
- subproducts doing "physical" GETMAIN requests, this enforces the Natural work pool allocation.

For more details concerning the ULIB RG and THABOVESIZE parameters, refer to the **Com-plete Utilities documentation**.

Support of Back-end Programs

Natural passes the following string to a back-end program:

- the Natural return code (fullword),
- the Natural termination message (A72),
- the length of the termination area (fullword),
- the termination data.

This string is mapped by the NAMBCKP macro.

The XNCFBACK source module is an example of a Natural back-end program in a Com-plete environment. It is written as reentrant code and can be loaded as RESIDENTPAGE program or once per user.

Com-plete Support in Natural Batch Runs

If you use the Com-plete services in a Natural batch run, the batch user ID remains logged on at the end of the batch run.

To avoid this situation, include the module COMPBTCH from the Com-plete distribution library in the batch Natural nucleus. This resolves the entry point for module EOJ, which is called at the end of the Natural batch job for termination clean-up.

The module NCFAM (previous name: NATCMPL) is used to access Com-plete print/work files. It has to be included in the linking of the Natural nucleus, together with the module COMPBTCH from the Com-plete distribution library.

Support of Asynchronous Natural Processing

Asynchronous Natural processing is discussed in the section Asynchronous Processing in the Natural Operations for Mainframes documentation; however, some additional considerations apply when running Natural under Com-plete.

Make sure that appropriate SENDER and OUTDEST destinations are specified for an asynchronous Natural session; otherwise, any output will lead to an abnormal termination.

An example to start an asynchronous Natural transaction under Com-plete can be found in the library SYSEXTP, program ASYNCOMP.

Invoking Natural from User Programs

The Com-plete FETCH function is used to invoke Natural from a user front-end program under Com-plete; see the **Com-plete Application Programmer's documentation** for details.

Storage Thread Key Handling

If you want to use protection mode between Com-plete and your application program, you must set the NATPARM parameter SKEY=OFF. The application program runs in the corresponding thread key. For any Natural or Editor buffer pool call, the front-end driver switches into the Com-plete key and back to the thread key after the call.

You can improve the performance of the application program dramatically under Com-plete Version 5.1.3 or higher by activating the Storage-Protection Override facility on your machine.

Set the thread key = 9 in the Com-plete startup parameter THREAD-GROUP for your Natural sub-group.

The front-end driver sets the Natural application automatically to the privileged mode if the thread key is 9, and uses the SPKA instruction for the key switch handling instead of using the Com-plete function MODIFY with function codes THRD/TCS.

Support of User Exit Handling during Session Initialization

During session initialization, it is possible to pass user-specific session information about the activation of a user exit to Natural. The exit is called before Natural has been initialized, after the driver/IOCB initialization is complete.

The driver passes as a parameter the address of the IOCB in register 1, whereas the exit is activated/deactivated by the Com-plete functions COLOAD/CODEL; see the **Com-plete Application Programmer's documentation** for details.

The NCFUEXIT source module is an example of a user exit. The user exit can be defined in the parameter module NCFPARM.

