Natural CICS Generation Parameters

This part of the Natural CICS Interface documentation describes the Natural CICS generation parameters.

The following topics are covered:

- NCISCPCB Generation Parameters

- NCMDIR Macro Parameters

- NCMTGD Macro Parameters

- NTSWPRM Macro Parameters

- NCIPAPM Generation Parameters

- NCMPRM Macro Parameters

References to CICS Tables

Where appropriate, any references to CICS tables (DCT, FCT, PCT, PPT, TCT, TST, etc.) can be considered as references to the corresponding:

- assembly-type resource definitions,

- online resource definitions via CEDA,

- batch resource definitions via DFHCSDUP.

Related Documents

- *Installing the Natural CICS Interface* in the Natural *Installation* documentation.

- Natural utility `SYSTP` which provides various TP-monitor-specific functions

- For information on operation and the individual components of Natural in a CICS environment, see the following sections in the *Operations* documentation:

  - *Node Error Program Considerations for Natural*

  - *CICS 3270 Bridge Considerations*

  - *Special Natural CICS Functionality*

  - *Natural CICS Sample Programs*

  - *NCIUIDEX User ID Exit Interface*

  - *Invoking Natural from User Programs*

  - *Asynchronous Natural Processing under CICS*

○ *Logging Natural Sessions under CICS*

○ *Performance Considerations*

○ *Natural CICS Interface Debugging Facilities*

○ *Natural Print and Work Files Under CICS*

NCISCPCB Generation Parameters

The Natural CICS interface system directory is generated by assembling and linking the `NCISCPCB` source module; see Installing the Natural CICS Interface, *Create the System Directory* in the Natural *Installation* documentation.

`NCISCPCB` contains the following macros:

- `NCMDIR`

- `NCMTGD`

- `NTSWPRM`

The purpose of these macros and the individual parameters which can be specified in the macros `NCMDIR` and `NCMTGD` are described in the following sections.

NCMDIR Macro Parameters

The `NCMDIR` macro is mandatory and must be specified as the first macro in the `NCISCPCB` source module. It contains various options for the system. The individual parameters which can be specified in the `NCMDIR` macro are described below.

`CICSPLX | ROLLSRV | SIPSERV | SUBSID | TSKEY | TSRECSZ | USERS`

CICSPLX - Switching of CICS Application Region

This parameter is applicable under z/OS only.

Possible values are:

Value:	Explanation:
YES	The Natural CICS interface keeps all session relevant data as the Session Information Records (SIRs) and the session data over a pseudo-conversational screen I/O outside of a local CICS Application Owning Region (AOR), thus enabling the switching of CICS AORs. Setting this parameter to YES also requires the profile parameter ADAMODE to be set to greater than 0.
MODE	This setting almost has the same meaning as YES; the only exception is that CICSPLX=MODE allows an ADAMODE=0 profile parameter specification, that is, CICS AOR switching is not possible, but a Natural session may survive the restart of a CICS AOR in an MRO environment.
NO	Vital Natural session data is kept in the local CICS AOR, which in fact disables CICS AOR switching. This is the default value.

Natural PLEX support means that a Natural CICS session removes all its footprints that exist in a CICS application region at CICS task end, as it might never come back into this region. Therefore all Natural CICS session relevant data must be kept outside of a CICS application region, that is, Natural under CICS passes its session information records (SIRs) to the *Authorized Services Manager*'s SIP handler and the session data to the *Natural Roll Server* at CICS task end. In addition to that, all modules 'held', that is, modules not linked to Natural but directly invoked via standard linkage conventions as RCA modules or the Adabas linkage module, have to be released at CICS task end. It also requires that the restart information is kept in a CICS terminal owning region (TOR) in case of COMARET=YES, or in a CICS data owning region (DOR), which is shared by all participating CICS AORs, in case of COMARET=NO, see the COMARET parameter for details.

If YES or MODE has been specified, and the NCMDIR SUBSID parameter has not been set, the value of the Natural profile parameter SUBSID in effect for the Natural session initializing the NCI environment will be taken.

> ⚠️ **Warning:**
> **Setting this parameter to YES or to MODE automatically sets**
> **SIPSERV and the ROLLSRV parameters to YES.**

ROLLSRV - Roll Server Rolling

This parameter is applicable under z/OS only.

Possible values are:

Value:	Explanation:
NO	This is the default value, if CICSPLX=NO and SIPSERV=NO. If CICSPLX or SIPSERV is YES, ROLLSRV=YES is forced.
YES	Specifying YES causes the Natural CICS interface to use the *Natural Roll Server* as roll facility only.

If the Natural Roll Server is to be used to save and restore the Natural session data over a screen I/O, this parameter must be set to YES, when the CICSPLX and SIPSERV parameters are both set to NO. If YES has been specified (or forced) and the NCMDIR SUBSID parameter has not been set, the value of the Natural profile parameter SUBSID in effect for the Natural session initializing the NCI environment will be taken.

Note that, for the purposes of the Natural CICS Interface, the Natural profile parameter SUBSID is only honored if it is specified dynamically or in the parameter module. It is ignored if it is specified in a parameter string by a profile parameter SYS or PROFILE or in an alternate parameter module (as specified with the profile parameter PARM).

SIPSERV - Authorized Services Manager's Session Information Pool

This parameter is applicable under z/OS only.

Possible values are:

Value:	Explanation:
NO	This is the default value, if CICSPLX=NO. If CICSPLX is not NO, SIPSERV=YES is forced.
YES	Causes the Natural CICS interface to keep its session information records (SIRs) in the *Authorized Services Manager*'s session information pool.

With this parameter set or forced to YES, the Natural session information records are kept outside a CICS region, thus enabling Natural to switch a CICS application region after a pseudo-conversational screen I/O.

If YES is specified (or forced) and the NCMDIR SUBSID parameter has not been set, the value of the Natural profile parameter SUBSID in effect for the Natural session initializing the NCI environment will be taken.

Note that, for the purposes of the Natural CICS Interface, the Natural profile parameter SUBSID is only honored if it is specified dynamically or in the parameter module. It is ignored if it is specified in a parameter string by a profile parameter SYS or PROFILE or in an alternate parameter module (as specified with the profile parameter PARM).

⚠ **Warning:**
If YES is effective for this parameter, the ROLLSRV parameter is forced to YES, unless already specified.

SUBSID - Sub-System ID

This parameter is applicable under z/OS only.

Possible values are:

Value:	Explanation:
SUBSID=*xxxx*	Defines the sub-system ID for the Natural Roll Server and/or for the Authorized Services Manager.

This parameter defines the Natural sub-system ID to be used for the Natural Roll Server and/or for the *Authorized Services Manager*. If this parameter is not specified, the value of the Natural profile parameter SUBSID will be taken.

Note that, for the purposes of the Natural CICS Interface, the Natural profile parameter SUBSID is only honored if it is specified dynamically or in the parameter module. It is ignored if it is specified in a parameter string by a profile parameter SYS or PROFILE or in an alternate parameter module (as specified with the profile parameter PARM).

TSKEY - Prefixes for Natural CICS Temporary Storage Key

This parameter defines the constant prefix of the temporary storage queue (see explanation below).

Possible values are:

Value:	Explanation:
(*xxxx*, *yyyy*)	TSKEY=(*xxxx*, *yyyy*) defines the prefix for roll data, whereas *yyyy* defines the prefix for pseudo-conversational restart data.
(NAT4,NCOM)	This is the default value.

When CICS temporary storage (main or auxiliary) is to be used for the Natural CICS interface roll facility or for the communication area for pseudo-conversational Natural tasks (as described with the NCMPRM macro parameter COMARET), names for queues of task dependent unique temporary storage must be specified.

These queue names consist of a constant 4-byte key and a task-related key. For terminal-dependent tasks, this task-related key corresponds to the terminal ID, for asynchronous non-terminal tasks it corresponds the CICS unique task number. The constant prefix of the temporary storage queue names is defined by the TSKEY parameter.

The Natural CICS interface requires two 4-byte prefixes: one for roll data and one for pseudo-conversational restart data. *xxxx* defines the prefix for roll data, *yyyy* defines the prefix for pseudo-conversational restart data. The two prefixes must be different from each other and exclusive for Natural under CICS.

When running in a CICSplex environment, the CICS temporary storage prefix for Natural session restart information must be defined in a CICS TST as REMOTE/SHARED to be accessible in all participating CICS regions.

TSRECSZ - Record Sizes for Main and Auxiliary Temporary Storage

This parameter defines the maximum record length for rolling of data if CICS temporary storage is to be used as Natural CICS interface roll facility.

Possible values are:

Value:	Explanation:
(*nnnnn,mmmmm*)	The first subparameter *nnnnn* applies to CICS main temporary storage and must be in the range of 4096 to 32763 or 0 or one of the keywords MAX, YES or NO; • if numeric non-zero, this value is used unconditionally; • if set to 0 or NO, CICS main temporary storage cannot be used for a Natural roll facility; • if set to MAX or YES, a record size of 32763 is taken. The second subparameter *mmmmm* applies to CICS auxiliary temporary storage and must be in the range of 3976 to 32763 or 0 or one of the keywords MAX, YES or NO; • if numeric non-zero, this value is used unconditionally; if set to MAX, a record size of 32763 is taken; • if set to NO, CICS auxiliary temporary storage cannot be used for a Natural roll facility; • if set to 0 or YES, the Natural CICS interface sets the record length which fits into an auxiliary temporary storage control interval, that is, CI size minus VSAM control information minus CICS control information. A user-defined record size greater than CI size results in fewer (logical) roll I/Os at the expense of additional CICS overhead due to writing spanned records.
(32748,0)	This is the default value.

USERS - Session Information Record

This parameter specifies the number of session information record slots (SIRs).

Possible values are:

Value:	Explanation:
(*nnnnn*, *mmm*)	The subparameter *nnnnn* defines the number of SIRs to be held in the Natural CICS directory module itself. *nnnnn* must be in the range from 1 to 32767. When the SIR slots in the directory are occupied, the Natural CICS interface acquires a CICS shared storage segment, large enough to hold the number of SIRs defined by *mmm*, which must be in the range from 0 to 255. If the subparameter *mmm* is 0 or omitted, the system does not acquire additional storage for SIRs if no free SIR slot is available in the system directory. If so, the Natural CICS system is actually restricted to the number of users specified by the first subparameter. If a value other than 0 is specified for *mmm*, secondary storage segments are allocated automatically as required. Allocated secondary segments are freed again if they are no longer needed.
(100,20)	This is the default value.

The Natural CICS interface permanently holds information about all active Natural sessions. Per session a so-called Session Information Record (SIR) is maintained.

These SIRs are kept

- in a Coupling Facility when running in a z/OS Parallel Sysplex environment;

- in a data space of the *Natural Authorized Services Manager* when running in multiple CICS regions inside a single z/OS system;

- in a CICS region's main storage when running in a single CICS AOR (locally).

However, whenever a Natural session is active in a CICS region, it will occupy a SIR slot in the current application region.

When running locally in a single CICS AOR, the USERS parameter applies to all Natural sessions. When running in a CICSplex environment, USERS applies to the subset of Natural sessions which is currently active in each of the participating CICS AORs.

NCMTGD Macro Parameters

The NCMTGD macro is mandatory and must be specified for each thread group. The Natural CICS interface allows you to define groups of threads. These groups are controlled/chosen by the CICS transaction ID at session initialization. The common thread size for the various groups may differ and the groups can have different options. The thread group definitions are part of the Natural CICS system directory, as they are relevant to the whole system, not just to a single session.

The individual parameters which can be specified in an NCMTGD macro are described below.

PFKEY | PRIMERF | THRDSZE | THREADS | TRAN | TYPE | XTRAN

PFKEY - PF/PA Keys for Thread Group

This parameter defines a single CICS transaction or a list of them.

Possible values are:

Value:	Explanation:
xxx	Possible values for *xxx* are: PF1 to PF24, PA1 to PA3.
(xxx,xxx,...)	Also a list of keys can be specified. This has to be enclosed in parantheses, for example, PFKEY=(PF12, PF14).

No default value is provided.

When starting a session, the Natural CICS interface scans through all thread group definitions for the current transaction ID, or PF or PA key. If it cannot be found, the first thread group is taken as default.

> ⚠ **Warning:**
> **At least one transaction ID (in character or hexadecimal format) or**
> **one transaction initiating attention identifier must be specified for all**
> **groups, except for the first group, which is used as the default group.**

PRIMERF - Natural CICS Primary Roll Facility

This parameter defines the Natural CICS interface primary roll facility for all tasks defined in the associated thread group. Therefore, this parameter does not apply to thread groups with TYPE=NONE.

Possible values are:

Value:	Explanation:
VSAM	The Natural CICS interface VSAM RRDS roll files are taken as the primary roll facility. CICS auxiliary temporary storage is considered as the secondary roll facility, which means that it is used if all primary roll files become full or unavailable.
AUX	CICS auxiliary temporary storage is taken as primary roll facility of the Natural CICS interface.
MAIN	CICS main temporary storage is taken as Natural CICS interface primary roll facility.
NONE	The associated sessions do not roll at all. NONE is not valid for TYPE=SHR groups and for groups with TYPE=ALIAS redefining TYPE=SHR groups.

No default value is provided.

This parameter is ignored when using the *Natural Roll Server*; if you force a Natural session with Roll Server to run conversationally with no rolling, value NONE is taken.

Points to be observed:

- PRIMERF=VSAM and PRIMERF=AUX have the same effect, when no VSAM RRDS roll file is available in the CICS system.

- PRIMERF=AUX and PRIMERF=MAIN have the same effect, when auxiliary temporary storage is not defined in the CICS system.

- If auxiliary temporary storage is not defined in the CICS system, a specification of PRIMERF=VSAM implies that CICS main temporary storage is considered as secondary roll facility, in case the VSAM RRDS roll files become unavailable or full.

- If CICS main temporary storage is to be used as roll facility, the record size is defined by the TSRECSZ parameter.

Important:
Note that sessions that are associated with thread groups defined with PRIMERF=NONE cannot roll due to the lack of a roll facility and are therefore conversational by design.

THRDSZE - Thread Size

This parameter defines the common thread size for TYPE=GETM and TYPE=SHR groups.

Possible values are:

Value:	Explanation:
nnn	The thread size *nnn* can be equal to 40 or greater.

No default value is provided.

Note that this parameter defines the *logical* thread size that is available to Natural. However, the Natural CICS interface NCI adds another 2 KB to the logical thread size for internal administration purposes. This means that the *physical* thread size or length of the thread GETMAIN request is by 2 KB greater than the THRDSZE value.

In case of TYPE=GETM, additional 16 bytes for the heading and trailing CICS storage accounting areas (SAAs) have to be considered.

Important Notes:

1. For GETMAINs of more than 512 KB, CICS aligns these storages at MB boundaries.

2. When using transaction isolation, CICS internally uses 1 MB "pages" in the EUDSA (see the *CICS Performance Guide for details*).

These two facts lead to storage fragmentation and should be kept in mind when setting an appropriate EDSALIM in CICS.

THREADS - Number of Threads or Tasks Per Thread Group

This parameter specifies the number of threads or tasks as described below.

Possible values are:

Value:	Explanation:
nnn	The number of threads can be equal to 510 or less.

No default value is provided.

For TYPE=SHR thread groups, the THREADS parameter is mandatory and defines the number of threads which are to be allocated via GETMAIN (SVC or SHARED, depending on CICS version) during installation.

For TYPE=GETM and TYPE=NONE thread groups, the THREADS parameter is optional and determines the maximum number of concurrently active Natural tasks per thread group. For these thread group types, the THREADS parameter does not control storage usage in contrast to TYPE=SHR thread groups (see also *Controlling Storage Usage*).

The number of threads or the number of tasks per thread group is defined by providing thread control blocks (TCBs).

While for TYPE=SHR thread groups, each thread is closely connected to its TCB. Threads are shared by queueing up on the associated TCB. Thread groups of TYPE=GETM and TYPE=NONE only queue up on a TCB to get active.

While sessions with TYPE=SHR thread groups compete for threads, the other session types compete for TCBs with a thread already allocated (TYPE=GETM) or with no allocated thread at all (TYPE=NONE).

When the THREADS parameter is non-zero, the Natural profile parameters DBROLL and MAXROLL and the calls to CMROLL are handled differently for TYPE=GETM/NONE thread groups: As threads cannot be released, the TCB resource held is released, which activates the session with the session data kept in storage.

TRAN - Transaction IDs for Thread Group

This parameter defines a single CICS transaction or a list of them.

Possible values are:

Value:	Explanation:
(see below)	One or more CICS transaction codes defined in the PCT for Natural.

No default value is provided.

The TRAN parameter expects transaction IDs to be in character format; transaction IDs with non-alphanumeric characters have to be enclosed in apostrophes.

When starting a session, the Natural CICS interface scans through all thread group definitions for the current transaction ID, or PF or PA key. If it cannot be found, the first thread group is taken as default.

A list of transaction IDs has to be enclosed in paranteses, for example, TRAN=(NATU, XYZ).

⚠️ **Warning:**
At least one transaction ID (in character or hexadecimal format) or
one transaction initiating attention identifier must be specified for all
groups, except for the first group, which is used as the default group.

TYPE - Thread Type for Group

This parameter defines which type of thread is to be used for a given group.

Possible values are:

Value:	Explanation:
SHR	CICS shared storage threads are used. The threads available for a thread group are shared by all CICS transactions defined for this group. Thread selection when starting a CICS task is done by an ENQUEUE/DEQUEUE technique. If currently no thread is available, a wait queue for this thread group is maintained. This is the default value. When running in a z/OS Parallel Sysplex environment, the Natural parameter RELO=OFF forces sessions with TYPE=SHR threads to be conversational to prevent a CICS region switch.
GETM	Threads allocated via GETMAIN are used, which means that a thread is actually acquired performing a CICS GETMAIN operation - EXEC CICS GETMAIN FLENGTH - with the thread group's common thread size. Using threads allocated via GETMAIN, each Natural task has exclusive thread storage available until it is terminated; that is, for pseudo-conversational tasks from screen I/O to screen I/O. If the Natural parameter RELO=OFF or PSEUDO=OFF is specified, tasks using threads allocated via GETMAIN are forced to be conversational, as there is no guarantee that after a FREEMAIN of the thread a subsequent GETMAIN obtains the same storage in memory. As thread storage allocated via GETMAIN exclusively belongs to the owning task, however, such tasks can be defined as non-rollable (see the PRIMERF parameter), which means that a given thread belongs to a given task until the end of the Natural session. If so, the task is conversational by design and no rolling is done.
NONE	No threads are used by transactions defined in this thread group and all Natural GETMAIN requests are directly passed to CICS for an EXEC CICS GETMAIN FLENGTH request. By design, such tasks cannot roll and are therefore conversational.
ALIAS	The current NCMTGD macro provides different options for the thread group defined by the previous NCMTGD macro specification. However, only thread groups of TYPE=GETM and TYPE=SHR can be redefined by one or more NCMTGD TYPE=ALIAS macro requests. Up to 99 thread groups are supported, which means that up to 99 NCMTGD macro specifications with TYPE other than ALIAS are recognized.

XTRAN - Hexadecimal Transaction IDs for Thread Group

This parameter is equivalent to the `TRAN` parameter, but it expects the transaction ID to be in hexadecimal format.

Possible values are:

Value:	Explanation:
(see below)	Possible values: one or more CICS transaction codes defined in the PCT for Natural.

No default value is provided.

A list of transaction IDs in hexadecimal format has to be enclosed in parantheses, for example, `XTRAN=(D5C1E3E4, E7E8E9)`.

> ⚠️ **Warning:**
> **At least one transaction ID (in character or hexadecimal format) or**
> **one transaction initiating attention identifier must be specified for all**
> **groups, except for the first group, which is used as the default group.**

NTSWPRM Macro Parameters

The `NTSWPRM` macro is used to define the various aspects of the swap pool. If no swap pool is to be used, omit this macro. For more information, see *Natural Swap Pool* in the Natural Operations documentation.

NCIPAPM Generation Parameters

The Natural CICS interface parameter module is generated by assembling the `NCIPARM` source module, see *Installing the Natural CICS Interface*, *Create the Natural CICS Interface Parameter Module* in the Natural *Installation* documentation. It holds the `NCMPRM` macro definition.

The purpose of the `NCMPRM` macro and the parameters which can be specified in it are described in the following section.

NCMPRM Macro Parameters

The macro `NCMPRM` determines all Natural session options that are relevant in a CICS environment. This macro is part of the Natural CICS parameter module, which is created in the corresponding step of the Natural under CICS *Installation Procedure* in the Natural Installation documentation.

A sample `NCMPRM` macro definition, including all default values, is contained in the `NCIPARM` source module in dataset `NCInnn.SRCE`.

The individual parameters of the `NCMPRM` macro are described below.

```
BACKEND | BACKOUT | CHAP | CNTCALL | COMACAL | COMAMSG | COMARET | CONSOLE | DIRNAME |
DUPTID | FLDLEN | LOGDEST | MSGDEST | MSGPFX | MSGTRAN | PREFIX | PRMDEST | PSTRNID |
RCVASYN | RESENDC | RESENDS | RJEDEST | RJEUSER | SIGNON | SLCALL | SLNOHLD | SNDLAST |
TERMVAR | TRANCHK | UCTRAN | UNITID
```

BACKEND - Back-End Program Invocation Control

This parameter defines whether a specified back-end program or transaction is to be invoked after the session has terminated (normally or abnormally).

The BACKEND parameter has two sub-parameters. The second sub-parameter is optional. It controls if a back-end program is to be invoked in the event of a terminal error. This also includes session clean-up tasks started by NEP.

Possible values are YES/NO for both sub-parameters, but the default values are different.

Value:	Explanation:
YES	Same as BACKEND=(YES,NO). This is the default if the BACKEND parameter is omitted. A potential back-end program or transaction is always invoked, particularly after task abends, but not in the case of terminal errors. When a back-end program is invoked, the Natural termination message and return code are passed to the CICS transaction work area (TWA). In addition, the same information can be passed to a CICS COMMAREA, as described with the COMAMSG parameter.
(YES,YES)	Same as BACKEND=(,YES). A potential backend program or transaction is always invoked, particularly after abends including terminal errors.
NO	Forces BACKEND=(NO,NO). A potential back-end program or transaction is only invoked if the Natural session has been terminated normally; that is, with a Natural termination message.

BACKOUT - Backout Transaction in the Case of Unrecoverable Abends

This parameter defines whether the Natural CICS interface is to perform a transaction backout by means of an EXEC CICS SYNCPOINT ROLLBACK call or not.

Possible values are:

Value:	Explanation:
YES	All pending file updates are backed out. This is the default value.
NO	All pending file updates are committed.

Because of its abnormal termination exit, the Natural CICS interface intercepts all abends. If an abend is not recoverable, all resources of the abending session are released and the session is terminated via EXEC CICS RETURN; that is, it is terminated "normally" in terms of CICS. Thus, at the end of the task, "pending" file updates are not automatically backed out by CICS.

CHAP - Change Task's Dispatching Priority

This parameter defines how the Natural CICS interface is to treat long-running tasks reaching the `DBROLL` and/or `MAXROLL` call limits. Possible values are:

Value:	Explanation:
YES	The task's dispatching priority is decremented by 1 every time it reaches the `DBROLL` and/or `MAXROLL` call limits. The original task dispatching priority is re-established at the next screen I/O.
NO	The session is suspended. This is the default value.

CNTCALL - CICS Call Passing Automatically Data in Container

With `SET CONTROL 'P=C'` the `CALL` statement parameter data is passed in a CICS COMMAREA on the `EXEC CICS LINK` rather than parameter data pointers. As a CICS COMMAREA is limited to 32 KB, `EXEC CICS LINK` with a COMMAREA greater than 32 KB will fail due to a LENGERR condition.

The `CNTCALL` parameter enables you to automatically use a container on `EXEC CICS LINK` when the data to be passed exceeds the maximum COMMAREA length of 32 KB. This functionality only works in CICS versions supporting channels and containers (CICS Transaction Server 3.1).

The default container name then is "NCI-COMMAREA" unless explicitly specified via the application programming interface `USR4204N` prior to the Natural `CALL` statement.

Possible values are:

Value:	Explanation:
YES	When the COMMAREA data would exceed 32 KB, the Natural CICS interface automatically uses a CICS container on the EXEC CICS LINK, using `NCI-COMMAREA` as default name.
NO	When the COMMAREA data would exceed 32 KB, the Natural `CALL` statement fails with a `NAT0920` message and reason code `LENGERR` (hexadecimal 16).

COMACAL - CICS COMMAREA Usage for Subroutine Calls

This parameter defines whether the Natural CICS interface is to take advantage of the CICS command level COMMAREA facility when invoking external subroutine programs with `EXEC CICS LINK`.

Possible values are:

Value:	Explanation:
YES	The Natural parameter list addresses are passed in both the CICS TWA (as with Natural Version 1) and in a CICS COMMAREA (as with Natural Version 2). The COMMAREA length setting is 8 or 12, depending on the FLDLEN parameter's setting. Specifying YES in particular improves the communication with external subroutine programs written in CICS command level, because it is easier for these subroutines to access a CICS COMMAREA than a CICS TWA; it also results in less overhead. This is the default value.
NO	Forces Natural to pass the Natural request parameter list address to an external subroutine program in the CICS TWA only. The COMMAREA length setting is 0.

Actually, the COMACAL parameter can provide "compatibility mode" to Natural Version 1 regarding the way of passing data to called external subroutines.

Set this parameter to YES if you want to take advantage of the call option SET CONTROL 'P=C'; see also the description of the terminal command %P in the *Terminal Commands* documentation.

COMAMSG - CICS COMMAREA Usage for Termination Messages

This parameter controls whether to pass the Natural termination message and return code to a potential back-end program or transaction in a CICS COMMAREA.

Possible values are:

Value:	Explanation:
YES	The Natural back-end parameter area and potential termination data are passed in a CICS COMMAREA. This is the default value.
NO	This setting forces Natural at session termination (normal or abnormal) to pass the Natural back-end parameter area (see also *Back-End Program Calling Conventions* in the Natural *Operations* documentation) to a potential back-end program in the CICS TWA only. Potential termination data are still passed in the COMMAREA. If there are no termination data available, no COMMAREA is passed.

COMARET - CICS COMMAREA Usage for Task Control

This parameter defines whether the Natural CICS interface is to take advantage of the CICS command level COMMAREA facility when terminating and restarting pseudo- conversational tasks.

Possible values are:

Value:	Explanation:
YES	A pseudo-conversational Natural task saves its restart information into a CICS COMMAREA, unless it has been invoked with `EXEC CICS LINK` or the equivalent CICS macro request. This is the default value.
NO	Forces Natural to place its restart information into CICS main temporary storage, which results in more overhead because of additional CICS service calls necessary to place and retrieve this information. The CICS temporary storage key used consists of a prefix string (as defined with the `NCMDIR` parameter `TSKEY` and of the terminal ID. If running in a CICSplex environment, the CICS temporary storage key prefix must be defined in a CICS TST as `REMOTE/SHARED` to be accessible in all participating CICS regions.

Actually the `COMARET` parameter can provide compatibility to Natural Version 1 in terms of where to put pseudo-conversational restart data.

CONSOLE - CICS Terminal ID for the Operator Console

This parameter specifies the terminal ID used by Natural for message switching to the operator console.

Possible values are:

Value:	Explanation:
xxxx	*xxxx* can be any 4-character terminal ID.
CN01	This is the default value under z/OS.
CNSL	This is the default value under z/VSE.

The console terminal must allow automatic transaction initiation (ATI) in the terminal entry.

DIRNAME - Name of Natural CICS Interface System Directory Module

This parameter specifies the name of the Natural CICS interface system directory module.

Possible values are:

Value:	Explanation:
(see below)	Any valid module name
prefix CB	*prefix* is the common prefix for programs and files, see `PREFIX` parameter. This is the default value.

The first 5 characters of the directory module name are also used as part of CICS temporary storage queue names related to the relevant NCI environment. So when running more than one Natural CICS environment in a CICS region, the relevant system directory module names must be different in the first 5 characters.

Note that the specified or defaulted Natural CICS interface system directory module name may be modified at run-time via the NCI system directory module name exit interface `NCIDIREX`. This makes it possible to use the same NCI driver/`NCIPARM`, but use different NCI environments (thread groups/thread sizes, etc.) depending for example on CICS system ID, transaction ID.

DUPTID - Handle Duplicate Terminal ID

The Natural CICS interface requires unique terminal IDs, because the terminal ID is the key for its session information records (SIRs). This is normally guaranteed for a single CICS region, but not necessarily over several CICS regions sharing the same SIP server.

The `DUPTID` parameter determines how the Natural CICS interface has to deal with duplicate terminal IDs, that is, when a new session is to be started and an SIR already exists for this terminal ID.

Possible values are:

Value:	Explanation:
YES	If a duplicate terminal ID is encountered, the Natural CICS interface internally forces the old session to terminate and, after that, starts a new session. This is the default value.
NO	When a SIR already exists for the new session's terminal ID, the Natural CICS interface terminates the new session and issues the message `NS19`. For an explanation and remedial actions, see *Natural under CICS Messages*, *SCP Processing Errors* in the *Messages and Codes* documentation.

A terminal ID exit interface is available to create unique 8-character terminal IDs, for example, by appending the 4-character CICS system ID to the physical 4-character CICS terminal ID, which results in a logical Natural terminal ID.

FLDLEN - Supply Field Length List On External Program Call

This parameter defines whether the field length list addresses are to be passed or not when invoking external subroutines via `EXEC CICS LINK`.

Possible values are:

Value:	Explanation:
NO	Same as FLDLEN=(NO,NO). Only the parameter address list address and the field description list address (R1 and R2, as described with the CALL statement) are passed in the CICS TWA and in a CICS COMAREA, unless COMACAL is set to NO. This is the default value.
YES	Same as FLDLEN=(YES,NO) The field length list address (R3, as described with the CALL statement) is passed in addition in a CICS TWA and in a COMMAREA, respectively.
(YES,YES)	Same as FLDLEN=(,YES) The field length list address and the large field length list address (R4, as described with the CALL statement) are passed in addition in a CICS TWA and in a COMMAREA, respectively. The second subparameter set to YES forces a YES condition for the first subparameter.

The following table lists the possible combinations of COMACAL and FLDLEN and their associated CALL parameter values:

NCIPARM Parameters		CALL Parameters		
COMACAL	FLDLEN	TWA parameter addresses	COMMAREA parameter addresses	COMMAREA length
NO	NO,NO	2	n/a	0
NO	YES,NO	3	n/a	0
NO	YES,YES	4	n/a	0
YES	NO,NO	2	2	8
YES	YES,NO	3	3	12
YES	YES,YES	4	4	16

In any case, the last address passed gets a flag saying it is the last address in the list. This flag is set in the high order bit in the address field.

LOGDEST - Natural CICS Logging Destination

This parameter specifies the name of a CICS destination, where the Natural CICS interface writes its session log records to.

Possible values are:

Value:	Explanation:
name	Any valid destination name
NLOG	This is the default value.

A CICS destination control table entry must be defined for the optional Natural CICS log dataset.

MSGDEST - Destination ID for Natural Error Message Logging

This parameter specifies the name of the CICS destination to be used by the Natural CICS interface for NCI informational messages and to log the Natural session termination message if a session terminates abnormally.

Possible values are:

Value:	Explanation:
name	Any valid destination name
NERR	This is the default value.

Since these messages are in character format, any already available CICS destination (for example, CSSL) can be used rather than defining a new one.

MSGPFX - Generate NCI Message Prefix for WTL Messages

The Natural CICS interface uses a prefix for all messages it sends to the MSGDEST destination. This prefix has a length of approximately 48 bytes and comprises the following information:

- NCI message number,

- CICS region SYSID,

- terminal ID or the string ASYN for non-terminal tasks,

- user ID,

- transaction ID,

- date and time.

By default, the message prefix is also appended to those messages which are output through CMWTL.

Possible values are:

Value:	Explanation:
YES	The `NCI` message prefix is appended to all messages which are issued through `CMWTL`. This is the default value.
NO	The `NCI` message prefix is not appended to the messages which are issued through `CMWTL`. The messages are issued unchanged.

MSGTRAN - Internal Message Switching Transaction ID

This parameter specifies the transaction ID internally used by the Natural message switching and asynchronous session flushing facilities.

Possible values are:

Value:	Explanation:
(see below)	Any valid CICS transaction ID.
NMSG	This is the default value.

This transaction ID must be different from any transaction ID used to invoke Natural, and it must be defined in CICS.

PREFIX - Common Prefix for Programs and Files

This parameter defines a common module prefix for the Natural CICS components as the Natural CICS system directory *prefix*CB, the CICS 3270 Bridge XFAINTU exit *prefix*XFA, the VSAM roll files *prefix*R*n*, *n* =1 - 9, and system control records in CICS main temporary storage holding information about all permanent GETMAIN storages by NCI as local pools and shared threads. The TS control record keys are of the form *prefix*X CR, where *X* is an unprintable character. It is anyhow good practice to use this common prefix for all NCI related programs, for example, *prefix*DRV for the Natural CICS interface nucleus, *prefix*NEP for the Natural CICS interface node error program.

Possible values are:

Value:	Explanation:
XXXXX	The *prefix* can be 1 to 5 bytes long and must conform to the naming conventions for programs and files.

No default value is provided.

PRMDEST - Name of the Natural CICS Profile Parameter Input Destination

This parameter specifies the name of a CICS destination containing Natural dynamic profile parameters.

Possible values are:

Value:	Explanation:
name	Any valid destination name
NPRM	This is the default value.

At system initialization time, the Natural CICS interface retrieves Natural dynamic profile parameters and saves them in its environment. At session start, potential other profile parameters (entered by way of terminal input or by a front-end caller) are concatenated at the end of the parameter string which was retrieved from the PRMDEST destination, that is, explicit dynamic profile parameters can be used to overwrite the Natural CICS interface system profile parameters read from PRMDEST.

A CICS destination control table entry must be defined for the optional Natural CICS interface profile parameter input destination, normally an extra partition destination.

PSTRNID - Control of *INIT-PROGRAM Variable Setting

When a Natural task is activated by a front-end program, the PSTRNID parameter determines, how the Natural system variable *INIT-PROGRAM is set.

Possible values are:

Value:	Explanation:
YES	*INIT-PROGRAM is set to the actual transaction ID used for Natural CICS pseudo-conversational task processing, which is not necessarily the transaction ID of the task which originally started the Natural session. This is the default value.
NO	*INIT-PROGRAM is set to the transaction ID of the task, which originally started the Natural session.

RCVASYN - Recover Asynchronous Session

This parameter defines how the Natural CICS interface treats asynchronous sessions.

Possible values are:

Value:	Explanation:
YES	This is the default value.
	The Natural CICS interface forces some Natural profile parameter settings for non-terminal sessions to prevent unexpected input or abends due to NT06, NT11 or other I/O errors.
	RCVASYN=YES forces the following parameter settings:
	• CM=OFF,MENU=OFF,PC=OFF
	• TTYPE=ASYL, if the SENDER specification is blank or not specified or the same as the NCIPARM MSGDEST specification.
	• SENDER='*msgdest*', if the SENDER specification is blank or not specified
	• OUTDEST='*sender*', if the OUTDEST specification is blank or not specified
NO	The Natural CICS interface does not do anything specific for non-terminal sessions; it is the user's responsibility to set appropriate Natural profile parameters for an asynchronous Natural session; see *Asynchronous Natural Processing*.

RESENDC - Check for Screen Re-sending after Subroutine Calls

Natural optimizes the 3270 output data stream by default. The screen imaging technique used by Natural makes it possible for Natural to always remember the map most recently sent. Thus, when sending a new map, Natural actually sends "updates" of the old map only. With this logic, a screen image can get destroyed by 3GL programs called by Natural which perform screen I/Os themselves.

Possible values are:

Value:	Explanation:
YES	The Natural CICS interface checks whether any called 3GL programs have performed screen I/Os. If so, the Natural CICS interface causes Natural to send a full screen with the next screen I/O.
	This is the default value.
NO	The Natural CICS interface causes Natural to send only updates.

RESENDS - Screen Re-send Check after Pseudo-Conversational Session Resume

Natural optimizes the 3270 output data stream by default. The screen imaging technique used by Natural makes it possible that Natural always remembers the map most recently sent. Thus, Natural only sends "updates" when sending a new map, too. With this logic a screen image can get destroyed, for example, by message switching (CICS CMSG transaction) during pseudo-conversational screen I/O.

Possible values are:

Value:	Explanation:
YES	During the Natural session, the Natural CICS interface also recognizes screen I/Os from outside and causes Natural to re-send the screen most recently issued. This is the default value.
NO	Natural only sends "updates" when sending a new map.

RJEDEST - Name of the Natural CICS Submit Destination

The parameter applies to z/OS-type operating systems only.

Possible values are:

Value:	Explanation:
(see below)	Destination name.
NRJE	This is the default value.

RJEDEST specifies the *destination name* of the CICS extra partition destination used by the NATRJE utility for submitting jobs via the JES internal reader facility.

> ⚠️ **Warning:**
> **An appropriate CICS destination must be defined in the CICS DCT and start-up JCL; see also the corresponding step of the *Installation Procedure for the Natural CICS Interface* in the Natural *Installation* documentation.**
> **Function code L or B (*parm3* of the NATRJE CALL statement) must be set for the last NATRJE call.**
> **When L is specified and *nrje* is an extra partition destination, the destination is closed, which in turn triggers the start of the internal reader.**
> **When B is specified and *nrje* is an indirect destination, the destination is not closed; in this case, a trailing /*EOF card must be submitted in order to trigger the start of the internal reader.**

For further information on the Natural NATRJE utility, refer to the Natural *Utilities* documentation.

RJEUSER - Submit to POWER User ID Setting

This parameter only applies to z/VSE operating systems using the POWER spooling system.

Possible values are:

Value:	Explanation:
YES	Sames as RJEUSER=(YES,CICS). This is the default value. The Natural system variable *INIT-USER is used as the XPCC user ID and the POWER JECL must be set up appropriately by the user.
(YES,NAT)	The Natural system variable *USER is used as the XPCC user ID and the POWER JECL must be set up appropriately by the user.
NO	The user ID 'R000' is used as the XPCC user ID for all jobs submitted by the Natural CICS interface.

In z/VSE operating systems, Natural under CICS performs job submission by means of XPCC macro requests.

The XPCC macro requires the specification of a user ID, thus giving access to the submitted job's list or punch output to the submitting user only, unless appropriate LDEST/PDEST parameters have been specified in the * $$ JOB statement or appropriate DEST parameters have been specified in the *$$ LST or *$$ PUN statement respectively.

Using the special user ID 'R000', however, gives common access to list or punch output of a submitted job without having to code appropriate target destinations in the JECL.

SIGNON - Signon Behavior

This parameter defines how Natural under CICS should deal with a CICS user ID for a Natural session.

Possible values are:

Value:	Explanation:
NO	This is the default value. Any non-blank value returned by EXEC CICS ASSIGN USERID (..) is considered to be valid.
YES	A non-blank value returned by EXEC CICS ASSIGN USERID (..) is considered to be valid if it is different from the CICS default user ID and if the user has signed on in CICS (terminal bound session only).

Further processing:

When a CICS user ID is invalid,

- the edited (unpacked) CICS task number is taken for non-terminal CICS sessions, that is, asynchronous or DPLed, etc.,

- for terminal bound sessions, the 3-byte CICS operator ID is taken when it is non-blank, otherwise the CICS terminal ID is taken.

Notes:

1. CICS terminal IDs are unique within a CICS region, while CICS user IDs and operator IDs are not necessarily. However, CICS terminal IDs may have duplicates in other CICS regions resulting in duplicate user IDs in Adabas.
2. Natural user ID exit `NATUEX1` or Natural CICS user ID exit interface `NCIUIDEX` may be used to customize `*INIT-USER`.

SLCALL - Standard Linkage Call

The Natural `CALL` statement invokes a dynamic non-Natural program using CICS conventions, that is, via an `EXEC CICS LINK`. A dynamic non-Natural program can also be invoked with standard linkage conventions (for example, `BALR/BASR/BASSM 14,15`) if an appropriate indicator is set in the Natural program before the `CALL` statement is executed; see also the terminal command `%P=S`, `%P=SC`, `%P=L` and `%P=LS`.

⚠️ **Warning:**
The terminal commands `%P=S`, `%P=SC`, `%P=L` and `%P=LS` bypass the
`SLCALL` automatism of using a certain linkage convention.

`SLCALL` enables you to automatically use a certain linkage convention. This is particularly relevant in CICS systems where the CICS macro level API is no longer supported, which is the case in CICS Version 3.2 or above and in all CICS/TS versions.

Possible values are:

Value:	Explanation:
YES	The Natural CICS interface determines whether the module to be called is a valid CICS command level program by looking for the string `DFH` at the module's load point. If `DFH` is found, the program is invoked via an `EXEC CICS LINK`. If `DFH` is not found, the module is treated according to standard linkage conventions and is invoked via `BALR/BASSM 14,15`.
NO	The linkage convention is not used. This is the default value.

SLNOHLD - Load Option for External Programs to Be Invoked via Standard Linkage Conventions

This parameter defines how the Natural CICS interface treats non-LE external programs to be invoked via standard linkage conventions (that is, dynamic non-CICS programs and RCA programs) in a non-CICSPlex environment.

Possible values are:

Value:	Explanation:
YES	This is the default value. The Natural CICS interface loads all non-LE external programs to be invoked via standard linkage conventions (including RCA programs) via `EXEC CICS LOAD` without the `HOLD` option, thus allowing these programs to be `NEWCOPY`ed while the Natural session is suspended/waiting in a pseudo-conversational screen I/O. `SLNOHLD=YES` corresponds to the processing which the Natural CICS interface does for LE programs in general and for non-LE programs in a CICSPlex environment anyhow.
NO	This is how Natural worked eversince. The Natural CICS interface loads all non-LE external programs to be invoked via standard linkage conventions (including RCA programs) via `EXEC CICS LOAD HOLD`, that is, such a program is fixed in storage for some time depending on the `DELETE` profile parameter setting, RCA programs until session end.

SNDLAST - LAST Option Usage for EXEC CICS SEND Commands

This parameter is useful for SNA terminals (`LUTYPE2`) with bracket protocol to force "end bracket" for pseudo-conversational screen I/Os.

Possible values are:

Value:	Explanation:
YES	The `LAST` option is used for `EXEC CICS SEND` commands before the task terminates in pseudo-conversational mode. This is the default value.
NO	The `LAST` option is not used.

TERMVAR - Terminal ID Variable for Natural Work Files

This parameter enables a Natural user to have exclusive Natural work files under CICS without having to know the terminal ID.

Possible values are:

Value:	Explanation:
xxxx	Variable *xxxx* is a four-character string. See explanation below.
&TID	This is the default value.

As terminal IDs are unique in a CICS system, exclusive work files in CICS temporary storage usually contain the CICS terminal ID. The parameter `TERMVAR` allows you to define a variable. If this variable is found in a work file name, it will be replaced by the actual terminal ID. Strings with non-alphanumeric characters must be enclosed in apostrophes (').

Note that for non-terminal sessions the packed CICS task number is used as a *logical* terminal ID.

> ⚠️ **Warning:**
> **The variable string must not contain the substring '**', because**
> **Natural will replace this substring with the work file number, which**
> **makes it impossible to insert the terminal ID.**

TRANCHK - Check Input Map for Transaction ID

If a connection to a CICS session gets lost or dropped (for example under VM or when a session manager is installed) without having terminated the session, another user can get into this open session when calling CICS. Usually, the first action of a user in a CICS environment is to enter a transaction ID.

Possible values are:

Value:	Explanation:
YES	The Natural CICS interface checks whether the first 4 bytes of the transaction ID entered by the user matches the Natural transaction ID. If so, the Natural CICS interface assumes a "restart" after a connection has been lost or dropped. All resources of the "old" session are freed and a new session is started.
NO	Data entered by the user are not checked for the Natural transaction ID. This is the default value.

UCTRAN - Lower/Mixed Case Support in Natural

This parameter enables or disables the lower/mixed case support by the Natural CICS interface (NCI).

Possible values are:

Value:	Explanation:
YES	Same as `UCTRAN=(YES,YES)`. NCI lower/mixed case support is fully enabled. This is the default value.
NO	Same as `UCTRAN=(NO,YES)`. NCI lower/mixed case support is disabled for pseudo-conversational screen I/Os.

The first subparameter controls NCI mixed case support after a pseudo-conversational screen I/O, while the second subparameter controls NCI mixed case support after a conversational screen I/O; the latter also includes NTC uploads.

First Subparameter (pseudo-conversational screen I/Os)

To accomplish lower/mixed case support for pseudo-conversational Natural sessions, it is necessary that the terminal input be not already translated to upper case before the Natural nucleus gets control. Therefore the Natural CICS interface by default switches terminals defined with `UCTRAN(YES)` into mixed mode (`UCTRAN(TRANID)`) for the lifetime of the Natural session.

As for security reasons any modification of CICS definitions/control blocks may not be desired, the Natural CICS interface can be prevented from modifying a terminal's upper case translation status by setting this NCIPARM parameter UCTRAN to NO. If so, the user must define a terminal as running in "lower case" (CICS TYPETERM parameter UCTRAN(TRANID/NO)) to be able to use the Natural lower/mixed case support.

As all CICS versions supported by the current Natural Version provide "case switching" on transaction level via the UCTRAN parameter in a transaction's PROFILE, this NCIPARM parameter should be set to NO, thus leaving lower/mixed case support to CICS.

Note:
In CICS, the combination of the UCTRAN parameters in both TYPETERM and PROFILE definitions determine how CICS treats the terminal input of a pseudo-conversational transaction (for details see CICS Resource Definition Manual or others). Therefor it is always advisable that mainly the PROFILE associated to a transaction defines the required upper case translation status thus making an application unaffected by any TYPETERM upper case translation mode changes.

Second Subparameter (conversational screen I/O)

Lower/mixed case support for conversational I/Os means that the Natural CICS interface uses the "as is" option on the CICS terminal input requests (CONVERSE/RECEIVE ASIS). If the second subparameter is set to NO, the Natural CICS interface does the conversational CICS terminal input requests without the "as is" option,

UNITID - Establish Unique Terminal IDs

This parameter helps to make the CICS terminal ID for Natural purposes unique over more than one CICS region.

Possible values are:

Value:	Explanation:
YES	The Natural CICS Interface appends a CICS system ID (local SYSID if no MRO, otherwise TOR SYSID) to the 4-byte CICS terminal ID, thus creating an 8-byte logical terminal ID.
NO	The Natural CICS Interface uses the CICS terminal ID as it is. This is the default value.

This parameter is of interest when resources are shared as SIP server or roll server by several CICS regions, particularly in non-CICSplex: If the same terminal IDs are used in several CICS environments, this parameter helps to provide unique terminal IDs for Natural. Inside the Natural CICS interface, Natural terminal IDs are 8-byte fields, and a combination of 8-byte terminal ID and 8-byte CICS user ID is taken as key for SIP and the roll server.

The result of this parameter is used by the Natural CICS Interface for the session key and the roll server key and by Natural for the system variable *INIT-ID.

Notes:

1. A terminal ID exit (`NCITIDEX`) possibly will post-process that logical terminal ID.
2. Also a user ID exit (`NCIUIDEX` and `NATUEX1`) may post-process the `*INIT-ID` system variable.
3. This parameter also applies to *Natural Advanced Facilities* (NAF) printers, that is, the printers have to be defined appropriately in the NAF spooling and report management system `NATSPOOL`, or a user ID exit should be used to post-process the `*INIT-ID` for printers.
4. For non-terminal sessions, the Natural CICS Interface always sets up an 8-byte logical *terminal ID* consisting of the packed CICS task number and the CICS system ID; that is, `UNITID=YES` is forced for asynchronous tasks with the CICS task number taken as terminal ID.