

Installing Con-nect SNADS Running Under CICS

This chapter describes the steps that must be performed to install Con-nect SNADS Version 1.5.1 under CICS.

It covers the following topics:

- Overview
- Installation Steps for CICS

Note:

To install Con-nect SNADS, Adabas and Natural DBA skills are required and you must be familiar with the Con-nect administration facilities as well.

Overview

Before you begin the installation process, you need the following information:

- The "connection" names used by the CICS system programmer when defining the adjacent nodes to CICS using CEDA for TCT generation (See step 11).
- The VTAM log mode names used by the CICS system programmer when defining sessions for the above mentioned connections.

Installation Steps for CICS

1. Load the Con-nect SNADS programs from data set *CSLnnn.INPL*

The *CSLnnn.INPL* data set contains a number of Natural programs that supplement the Con-nect system program in the SYSCNT2 application. The Natural utility program INPL must be used to load the contents of the data set into the appropriate Natural system file. Objects are loaded only into the SYSCNT2 system libraries.

2. Use the local macro libraries and the CICS preprocessor, *DFHEAP1\$*, to assemble and link-edit *CXLU62* if not already done for transport service

CXLU62 is the Con-nect external communication module and is located on the *CNTnnn.SRCE* data set. The *CXLU62* module is written in Assembler and uses CICS commands.

- Process the *CXLU62* communication module using the CICS command language processor *DFHEAP1\$*.
- Assemble the translated output using the appropriate CICS, Natural, and Natural CICS source data sets as macro libraries.
- Link-edit the *CXLU62* module. The *CXLU62* module is referenced by the *CSTATIC* list of a Natural parameter module. Since a *BALR* instruction must be used to branch to the *CXLU62* module (rather than by a CICS *LINK* or *XCTL* command), the system administrator can use one of the following three options to link-edit the *CXLU62* module:
 - Option One: The *CXLU62* module is referenced by the *CSTATIC* list from a Natural parameter module that is part of a Natural CICS nucleus, and *CXLU62* is linked to the Natural CICS nucleus. In this case, the *CXLU62* module uses the CICS exec interface stub included with the Natural CICS nucleus.

With this option, the *CXLU62* module should not be linked with the CICS exec interface stub modules.

- Option Two: The CXLU62 module is referenced by the CSTATIC list from a Natural parameter module that is part of a Natural CICS nucleus, and CXLU62 is not linked to the Natural CICS nucleus. In this case, the CXLU62 module is loaded by CICS each time a Natural session is initialized by using the Natural RCA (Resolve CSTATIC Addresses) feature. The setting RCA=ON is used to activate the RCA parameter.

With this option, the CXLU62 module must be linked with the CICS exec interface stub modules. Since Natural will branch to the beginning of the CXLU62 load module, the CSECTs of the load module must be linked in the following sequence: CXLU62, DFHEAI, and then DFHEAI0. The name of the resulting load module must be CXLU62.

Next, the CXLU62 module must be copied into one of the step libraries of the CICS nucleus and referenced by the appropriate PPT entry. The PPT entry must indicate that CXLU62 is resident. Additionally, if you are running CICS/ESA 3.3 or above and this module is intended to be used in 31-bit addressing mode, it is recommended that DATALLOCATION=ANY is specified in the PPT entry.

- Option Three: The CXLU62 module is referenced by the CSTATIC list from a Natural parameter module that is not part of a Natural CICS nucleus, but is, nevertheless, loaded at Natural session initialization by the Natural keyword PARM.

With this option, both the CXLU62 module and CICS exec interface stub modules must be linked with the Natural parameter module. Because Natural assumes the parameter module referenced in the PARM keyword is located at the beginning of the load module, the CSECTs that comprise the Natural parameter module must be located at the beginning of the resulting load module.

Next, the module must be copied into one of the step libraries of the CICS nucleus and referenced by the appropriate PPT entry. The PPT entry must indicate that the module is resident. Additionally, if you are running CICS/ESA 3.3 or above and this module is intended to be used in 31-bit addressing mode, it is recommended that DATALLOCATION=ANY is specified in the PPT entry.

- The linkage editor may flag a few unresolved external references when you link-edit a Natural parameter module that is designed to be loaded dynamically. This is unavoidable and will not cause any problems. If CICS refuses to load a module that has been marked as "not executable", specify either the linkage editor NCAL or the LET option.

3. Relink your Natural nucleus (or nuclei) to satisfy the requirements for NATPARMs, the interface module CXLU62, and the driver module NATTTY

Certain requirements must be satisfied when generating the Natural nucleus (or nuclei) to be used by the "end user" and "server programs" applications of Con-nect SNADS. The requirements for the "end user" application are less stringent than the requirements for the "server programs" application.

Following are the requirements for the Con-nect SNADS "end user" application:

- The environment must include a fully operational Con-nect system Version 3.4.1.
- The environment must contain a valid reference to a Con-nect spool file. The spool file identifier number is 223. This reference can be specified in the Natural parameter module with a NTLFILE macro call, or dynamically using the Natural keyword LFILE.

Note:

Do not use Adabas passwords or cipherring for either the Con-nect spool file or the Con-nect system file.

- The Con-nect SNADS system programs must be loaded into the appropriate Natural system file.
- The Natural keyword CSIZE which specifies the size of the Con-nect buffer, must be assigned a value of 2 (this is the minimum value that can be assigned).

Note:

The following requirements only apply if event-driven scheduling is used for the Con-nect SNADS inbound queue; that is, if the DS_ROUTER_DIRECTOR program is to be immediately invoked when a send request is submitted or when the DS RECEIVE program receives a SNADS message from another node within the network.

- The Natural nucleus must run in a CICS environment.
- The CSTATIC list from the Natural parameter module must specify the module name CXLU62.
- The CXLU62 module must be either linked to or accessible to the Natural nucleus.
- Following are the requirements for the Con-nect SNADS "server programs" application:
- All of the requirements for the "end user" application also apply to the "server programs" application.
- The Natural system keywords listed below must be set as follows:

Natural Keyword	Required Setting
WH	ON
PSEUDO	OFF
RELO	OFF
MADIO	0
MAXCL	0
AUTO	OFF
ETID (see note below)	Blank

Note:

The setting ETID=' ' (blank) is recommended but not required. However, if more than one instance of a Con-nect SNADS server program becomes active at the same time, and more than one of those instances uses the same Natural user ID, Adabas will not issue an error message if the Natural keyword ETID is set to blank.

4. Modify the "queue server" front-end parameter module CONSNADS

Note:

Always check that the CONSNADS parameters are valid. If not, you have to re-execute this and the following three steps.

The CONSNADS module is an Assembler program that uses the keyword macro CSNADS. It is linked to the four Con-nect SNADS "queue server" front-end programs in order to provide the information required to handle the Natural front-end logic, e.g. the name of the resident Natural CICS nucleus, and to specify the dynamic parameters that must be passed to the Natural sessions.

Natural parameters that were not included as NATPARMs for the nucleus can be specified dynamically in the CONSNADS parameter module.

- In the CONSNADS parameter module, you must specify:
 - the names used for each of the four Con-nect SNADS front-end programs. The supplied front-end programs are: CSINIT, CSRECV, CSSEND, and CSTERM;
 - the names used for each of the four Con-nect SNADS front-end programs. The supplied front-end programs are: CSINIT, CSRECV, CSSEND, and CSTERM;
 - the Natural nucleus used for the "server programs" application; and
 - the appropriate values for the CONSNADS keyword parameters that are used by the CSNADS macro (see the table below).
- Assemble the CONSNADS parameter module using the Con-nect SNADS source data set as a macro library.

Required CONSNADS Keyword Parameters

The following table lists the required CONSNADS keyword parameters and describes the value associated with each one:

Keyword	Value
DYNINIT	A dynamic parameter string in a Natural stack that is passed to the Natural nucleus in order to execute the "initialize queues" program. The parameter string must contain the Natural commands required to log on to the SYSCNT2 application, execute the command "X-FMRUN I" (if you are using the watchdog function, replace the command "X-FMRUN I" with "X-FWTCH %"), and terminate the Natural session.
	Example stack parameter: <code>STACK=(LOGON SYSCNT2;X-FMRUN I;FIN)</code>
	In addition, the DYNINIT parameter string must satisfy the NATPARM requirements for the "server programs" application, e.g. WH=ON.
DYNRECV	A dynamic parameter string in a Natural stack that is passed to the Natural nucleus in order to execute the DS_RECEIVE program. The parameter string must contain the Natural commands required to log on to the SYSCNT2 application, execute the command "X-FIX000", and terminate the Natural session.
	Example stack parameter: <code>STACK=(LOGON SYSCNT2;X-FIX000;FIN)</code>
	If the Natural keyword ETID is not assigned a value of blanks, the user identifier must be represented in the parameter string by a dollar sign character (\$).
	Example stack parameter: <code>STACK=(LOGON SYSCNT2,\$,PASSWORD;X-FIX000;FIN)</code>
	In addition, the DYNRECV parameter string must satisfy the NATPARM requirements for the "server programs" application, e.g. WH=ON.
DYNROUT	A dynamic parameter string in a Natural stack that is passed to the Natural nucleus in order to execute the DS_ROUTER_DIRECTOR program. The parameter string must contain the Natural commands required to log on to the application SYSCNT2, execute the command "X-FRP", and terminate the Natural session.
	If the Natural keyword ETID is not assigned a value of blanks, the user identifier must be represented in the parameter string by a dollar sign character (\$).
	Example stack parameter: <code>STACK=(LOGON SYSCNT2,\$,PASSWORD;X-FRP;FIN)</code>
	In addition, the DYNROUT parameter string must satisfy the NATPARM requirements for the "server programs" application, e.g. WH=ON.

Keyword	Value
DYNSEND	A dynamic parameter string in a Natural stack that is passed to the Natural nucleus in order to execute the DS_SEND program. The parameter string must contain the Natural commands required to log on to the application SYSCNT2, execute the command "X-FOX000 %", and terminate the Natural session. The percent sign (%) is substituted at execution time by the identifier that represents the queue to be processed.
	If the Natural keyword ETID is not assigned a value of blanks, the user identifier must be represented in the parameter string by a dollar sign character (\$).
	Example stack parameter: <code>STACK=(LOGON SYSCNT2,\$,PASSWORD:X-FOX000 %;FIN)</code>
	In addition, the DYNSEND parameter string must satisfy the NATPARM requirements for the "server programs" application, e.g. WH=ON.
DYNTERM	A dynamic parameter string in a Natural stack parameter that is passed to the Natural nucleus in order to execute the "terminate queues" program. The parameter string must contain the Natural commands required to log on to the application SYSCNT2, execute the command "X-FMRUN T", and terminate the Natural session.
	Example stack parameter: <code>STACK=(LOGON SYSCNT2;X-FMRUN T;FIN)</code>
	In addition, the DYNTERM parameter string must satisfy the NATPARM requirements for the "server programs" application, e.g. WH=ON.
INITPGM	The name of the front-end program, e.g. CSINIT, used to invoke the "initialize queues" function. Make a note of the value you assign as you will need it when defining the CICS PPT.
NPGRMID	The name of the resident Natural CICS nucleus previously defined in the CICS PPT. If the "OS Core" load option is used, this must not be the same name as the Natural CICS nucleus load module.
NTRANID	The name of the CICS transaction code previously assigned to the Natural CICS nucleus in the CICS PPT.
RECVPGM	The name of the front-end program, e.g. CSRECV, used to invoke the DS_RECEIVE program. Make a note of the value you assign as you will need it when defining the CICS PPT.
SENDPGM	The name of the front-end program, e.g. CSSEND, used to invoke the DS_SEND and DS_ROUTER_DIRECTOR programs. Make a note of the value you assign as you will need it when defining the CICS PPT.
TERMPGM	The name of the front-end program, e.g. CSTERM, used to invoke the "terminate queues" function. Make a note of the value you assign as you will need it when defining the CICS PPT.

Optional CONSNADS Keyword Parameters

The following table lists the optional CONSNADS keyword parameters and describes the value associated with each one.

Note:

These parameters are available with Con-nect SNADS 1.5.1, but may be withdrawn in future versions.

Keyword	Value
RNTASKS	A value from 1 through 99 which represents the number of user identifiers available to Con-nect SNADS. This keyword applies only to the DS_RECEIVE program, and if the value of the Natural keyword ETID is not blank.
RTRYCNT	A value from 0 through 32767 which represents the maximum number of attempts that can be made to obtain an available user identifier. If the value of RTRYCNT is exceeded, the respective Con-nect SNADS server program will abend (code CS00).
	A value of 0 indicates that no attempts can be made. If no value is specified, 3 is the default.
	This keyword applies only to the DS_RECEIVE program and only if the value of the Natural keyword ETID is not blank.
RTRYDLY	A value from 1 through 32767 which represents the number of seconds between attempts to obtain an available user identifier.
	If no value is specified, 20 is the default.
	This keyword applies only if the value of the Natural keyword ETID is not blank.
RUIDPFX	A one-to-six character string used as the prefix for user identifiers. This value is concatenated with a two-digit number which has a value from 1 through 99 to form a user identifier.
	Note: The value specified for the RUIDPFX parameter must not be the same value specified for the SUIDPFX parameter.
	If no value is specified, the character string RSNADS is the default.
	This keyword applies only to the DS RECEIVE program and if the value of the Natural keyword ETID is blank.
SNTASKS	A value from 1 through 99 which represents the number of user identifiers available to Con-nect SNADS.
	This keyword applies only to the DS_SEND and DS_ROUTER_DIRECTOR programs, and if the value of the Natural keyword ETID is blank.

Keyword	Value
SUIDPFX	A one-to-six character string used as the prefix for user identifiers. This value is concatenated with a two-digit number which has a value from 1 through 99 to form a user identifier.
	Note: The value specified for the SUIDPFX parameter must not be the same value specified for the RUIDPFX parameter.
	If no value is specified, the character string SSNADS is the default. This keyword applies only to the DS_SEND and DS_ROUTER_DIRECTOR programs, and if the value of the Natural keyword ETID is not blank.

5. Specify keyword parameters (if required) to perform periodical supervision ("watchdog") of the Con-nect SNADS tasks

The watchdog program is generated to periodically supervise the status of the system and attempt to recover it automatically when necessary. When a temporary system failure is detected, the watchdog program issues a message to the operator console and records the event in the Con-nect SNADS log records. Then the watchdog program re-adjusts the status queues and re-starts the queue servers. However, the watchdog program cannot acquire inactive APPC connections.

The following table lists the required keyword parameters and describes the value associated with each one:

Keyword	Value
WATCH	A value of YES or NO. When the value is YES, the watchdog program is generated and replaces the Initialize Queues function. The default value is NO.
WTRANID	The CICS transaction code which is assigned to the CSINIT program.
WATCHIN	The length of time (in minutes) between subsequent executions of the watchdog program. The default value is 15.
WATCHRQ	A one-to-eight character string used by the watchdog. The watchdog program occupies one CICS request ID, which is specified here. The request ID specified must not conflict with any other CICS request ID.
	The default value is WATCH.

If the watchdog program is generated, apply the following modification to the CONSNADS keyword parameter DYNINIT:

Replace the command "X-FMRUN I" with "X-FWTCH %".

6. Assemble and link (if required) the Con-nect SNADS Adabas interface module

Use the default options PROLOG and EPILOG of the CICS command language translator DFHEAP1\$.

The Con-nect SNADS Adabas interface module is a subroutine which permits Adabas OPEN (OP command) and CLOSE (CL commands) commands to be executed from within the front-end programs. The interface module is a member of the source data set CSDBAS.

It is not necessary to use the Adabas interface module unless your Con-nect SNADS front-end programs are to issue OP and CL commands. The functionality can be useful if the value of the Natural keyword ETID is set to blank.

If the Adabas interface module is entered via the CICS LINK command, CSDBAS must be assembled and link-edited as a standard CICS application program (i.e. with the CICS exec interface stub modules DFHEAI and DFHEAI0). CONSNADS must be included when link-editing CSDBAS. A PPT entry is required in this instance and must indicate that the module is resident. The name of the PPT must match the value specified as ADAPGM. If CICS/ESA 3.3 or above is installed and this module is intended to be used in 31-bit addressing mode, it is recommended that DATALLOCATION=ANY is specified in the PPT entry.

If OP and CL commands are to be issued by the front-end programs and the Con-nect SNADS Adabas interface module is not entered via the CICS LINK command, CSDBAS must be link-edited with each of the front-end programs. In this case, the CICS exec interface stub modules and the PPT entry are not required.

The following table lists the required keywords and describes the value associated with each one:

Keyword	Value
ADADBID	Specify the database number which OP and CL commands are to be issued against. If files in multiple databases are to be opened and closed, the database numbers must be specified in subparameter notation. For example the OP and CL commands will be directed to database 2: ADADBID=2
	The OP and CL commands will be directed to databases 10 and 15: ADADBID=(10 , 15)
ADAUID	Specify whether the startup programs will generate an Adabas user ID when issuing OP commands. Usually the intention of using ETID=' ' (blank) is not to use Adabas user IDs. Therefore this keyword should not be specified.
	The default value is NO.
ADANAME	The name of the Adabas Link module in your CICS environment. The default value is Adabas.
ADALINK	Specify whether the Con-nect SNADS Adabas interface module is entered via the CICS LINK command (in which case the interface module will occupy a PPT entry of its own) or if it is link-edited with each of the startup programs.
	The default value is NO. Change the default value to YES if you want the module to be entered via the CICS LINK command.
ADAPGM	The name of the Con-nect SNADS Adabas interface module.
	The default value is CSDBAS.
ADAOPRB	Specify the record buffer contents for the OP commands. If this parameter has not been specified, the OP commands are not issued.
	If OP commands are to be directed to multiple databases, multiple record buffer contents must be specified in subparameter notation. The number of all Natural and Con-nect system files in use must be specified. For example the OP commands will be directed to database number 2 with the record buffer contents 'UPD=4,8,12.': ADADBID=2 , ADAOPRB='UPD=4 , 8 , 12 . '
	The OP commands will be directed to databases 5 and 15 with the record buffer contents 'UPD=3,6.' and 'ACC.'. OP commands will not be directed to database 10: ADADBID=(5 , 10 , 15) , ADAOPRB=('UPD=3 , 6 . ' , , 'ACC . ')
ADACLSE	Specify whether the startup programs will issue CL commands at the end of the session.
	The default value is NO.

7. Assemble and link-edit the "queue server" front-end programs

The names of the supplied front-end programs for the "queue server" functions are as follows:

Program	Queue Server Function
CSINIT	Initialize queues
CSRECV	DS_RECEIVE
CSSEND	DS_SEND and DS_ROUTER_DIRECTOR.
CSTERM	Terminate queues

The CSSEND and CSRECV front-end programs are required. Depending on how you have defined the DIU queues within the Con-nect Administration facility, the CSSEND and CSRECV programs are started either asynchronously at specified time intervals, or when a particular event occurs.

The CSINIT and CSTERM front-end programs are optional. CSINIT is invoked when CICS is initialized and CSTERM is invoked when CICS is terminated. If these front-end programs are not generated, queue server activity is activated when an event occurs or at specified time intervals, unless the Con-nect system administrator uses the Con-nect ADMIN program to terminate or initialize the queue server activity.

Note:

The names of the load modules created in this step must correspond to the values specified for the INITPGM, RECVPGM, SENDPGM, and TERMPGM keywords of the CONSNADS parameter module.

- Use the CICS command language translator DFHEAP1\$ with the default options PROLOG and EPILOG to process the queue server front-end programs.
- Assign the appropriate CICS source data set and the Con-nect SNADS source data set as macro libraries, and assemble the queue server front-end programs.
- Link-edit the queue server front-end programs with the CICS exec interface stub modules DFHEAI and DFHEAI0, and the CONSNADS parameter module created in the previous step. When link-editing the queue server front-end programs, place the CICS exec interface stub program DFHEAI first.
- Copy the queue server front-end programs into an appropriate step library of the CICS procedure.

8. Update the CICS PPT and PCT using CEDA (the CICS resource definition transaction)

- Define the queue server front-end programs in the CICS PPT and mark them as resident.

The name you specify for the queue server front-end programs in the CICS PPT must be the same as the values specified in the INITPGM, RECVPGM, SENDPGM, and TERMPGM keywords of the CONSNADS parameter module. If CICS/ESA 3.3 or above is installed and these programs are intended to be used in 31-bit addressing mode, it is recommended that `DATALOCATION=ANY` is specified in the PPT entry.

- Assign a CICS transaction code to each of the queue server front-end programs in the CICS PCT. The size of the Transaction Work Area (TWA) must be a minimum of 256 bytes. The actual TWA space requirement is dependent on the length of the dynamic parameter string specified in the CONSNADS parameter module. In most cases, a TWA size of 512 bytes is adequate. If the TWA size is not adequate, the queue server front-end programs will abend (code CS01). If CICS/ESA 3.3 or above is installed and these programs are intended to be used in 31-bit addressing mode, it is recommended that `TASKDATALOC=ANY` is specified in the PCT entry.
- Assign an external transaction identifier (XTRANID), in the appropriate PCT entry for the front-end programs used with the `DS_RECEIVE` program and the `DS_SEND` and `DS_ROUTER_DIRECTOR` programs. The external transaction IDs that must be used are:
 - 21F0F0F1 for the queue server front-end program that invokes the `DS_SEND` and `DS_ROUTER_DIRECTOR` programs;
 - 21F0F0F2 for the queue server front-end program that invokes the `DS_RECEIVE` program.

When the values 21F0F0F1 and 21F0F0F2 are entered, the CICS CEDA utility may issue a warning indicating that these transaction codes are reserved for IBM use. This message can be ignored.

9. Define local node for each Con-nect spool file serviced

Note:

If you change the DBID or FNR of a Con-nect system file, you have to re-execute this and the following step.

Con-nect SNADS uses spool file method routines for transferring data from the Con-nect system file (for local node data) to the Con-nect spool file (for external node data) and vice versa. Thus, a local node for the spool file method must be defined.

- Select the System Maintenance function from the Con-nect Administration Menu.
- Invoke the External Mail Nodes function to display the "Administration - External Mail Nodes" screen.



Warning:

The following sub-step must be completed before you begin step 10.

- Select the Define Local Node function and, if you have not already done so, define a name for your spool file in the "Define Local Node" field and press ENTER. You are then prompted to define a local node for the transport service, although it is not required for Con-nect SNADS. Define a name in the "Define Local Node" field and press ENTER. For further information, see the *Con-nect Administration* documentation, section *External Mail Nodes*, sub-section *Define Local Node*.

By default, Con-nect SNADS uses the name you enter in the "Define Local Node" field for the spool file method as the Distribution Group Name (DGN) when mapping local Con-nect user identifiers to the SNADS Distribution User Names (DUNs).

10. Define control information for nodes (system files) - SNADS Initialization

You must define the control information for all SNADS distribution service units (DSUs) that will be serviced from the current Con-nect spool file as local SNADS nodes.

To define Con-nect system files as nodes in the SNADS network, perform the following steps:

- Select the System Maintenance function from the Con-nect Administration - Main Menu.
- Invoke the External Mail Nodes function to display the "Administration - External Mail Nodes" screen.
- Select the Maintain Mail Nodes function and mark "F Snads" in the resulting Maintain Mail Node window.
- Select the Initialization function from the SNADS Administration Menu to display the "Control Maintenance" screen.
- On the this screen, specify the DSUN (RGN and REN) of the Con-nect system file. In this context, a "Con-nect system" is the system file.

For more information about the "Control Maintenance" screen refer to the *Con-nect Administration* documentation, section *Maintaining Con-nect SNADS*, sub-section *Control Maintenance*.

11. Create one or multiple outbound queues for the storing of Distribution Interchange Units (DIUs) intended for other nodes

Note:

This and the following three steps define the path messages take from Con-nect to other nodes. If changes are made in your SNA network, you may have to re-execute this step.

At least one outbound queue must be created for each adjacent node in the SNADS network. When an outbound queue is created, the attributes of the LU6.2 link between CICS and the adjacent SNADS node, e.g. a Soft-Switch node, must be defined. To create an outbound queue, perform the following steps:

- Select the Queue Maintenance function from the SNADS Administration Menu. The "Queue Maintenance" screen is displayed. Press PF4 to add a queue. The "Queue Info" screen is displayed.
- In the this screen, enter the following:
 - In the "Queue ID" field, enter a one-to-eight character name to be used as the queue ID.
 - In the "Queue ID" field, enter a one-to-eight character name to be used as the queue ID.
 - In the "Connection ID" field, enter the connection name used by the CICS system programmer to define the adjacent node to CICS.
 - Enter the log mode name that the CICS system programmer designated to be used with the connection specified above.
 - Press PF5 (Confirm) to add the new outbound queue.

For further information about the "Queue Info" screen, refer to the *Con-nect Administration* documentation, section *Configuring Con-nect SNADS*, sub-section *Outbound Queues*.

12. Set the "Reset" and "Input" statuses for the outbound queue

When you initially test Con-nect SNADS, the input status of the outbound queue must be set to active, and the reset status to inactive. To set the input and reset statuses of the outbound queue, perform the following steps:

- Select the Queue Maintenance function from the "SNADS Administration Menu" screen to display the "Queue Maintenance" screen.
- Enter MO (modify) in the "Cmd" column for the outbound queue and press ENTER. The "Queue Info" screen is displayed. Modify the following fields:
 - In the "Reset Status" field, enter I (inactive).
 - In the "Input Status" field, enter A (active).
 - Mark the "Reset" field with any character and press PF5.

For more information about the "Queue Maintenance" screen refer to the *Con-nect Administration* documentation, section *Maintaining Con-nect SNADS* , sub-section *Queue Maintenance*.

13. Define routing specifications for all other nodes with which you intend to communicate

You must define a routing entry for each node in the network with which you wish to communicate. The routing entry determines how the SNADS DIUs are to be routed to a particular node.

To define routing specifications, perform the following steps:

- Select the Routing Entry Maintenance function from the SNADS Administration Menu to display the "Routing Entry Maintenance" screen. Press PF4 to add a routing entry. The following must be specified in the "Routing Entry" screen:
 - In the "Recipient Node" field, enter the DSUN (RGN.REN) for each node.
 - In the "Next Queue" field, enter the name of an outbound queue defined in step 12.
 - Press PF5 (Confirm) to add the new routing entry.

For more information about the "Routing Entry" screen, refer to the *Con-nect Administration* documentation, section *Maintaining Con-nect SNADS*, sub-section *Routing Entries*.

14. Add a mail node, type F (SNADS), for sending purposes

You must define an external mail node in Con-nect for sending mail.

To define an external mail node, perform the following steps:

- Select the System Maintenance function on the Con-nect Administration - Main Menu to display the "Administration - System Maintenance" screen.
- Select the External Mail Nodes function and press ENTER to display the "Administration - External Mail Nodes" screen.
- Specify the node name in the "Mail Node/Type" field (this node name will be used on the "Addressee" field in the Con-nect "Send" screen), select the Add Mail Node function and press ENTER.
- Enter F (SNADS) in the "Node Type" field.
- Enter the RGN (Group) and REN (Element) for each external SNADS node (i.e. addressee) to form the DSUN (Node) portion of the SNADS address.

When an object (document or memo) is sent to this addressee, the value you entered for the DSUN (RGN and REN) is automatically provided by the system. The user must supply the DUN (DGN and DEN) for a specific user in order to complete the SNADS address.

15. Test the installation of Con-nect SNADS

When testing, use the following settings for ALL queues:

Input Status: A

Output Status: I

Reset Status: I

This guarantees that the queue servers will be activated only when they are started from the "Queue Maintenance" screen.

If you try to start a queue server when either the outbound or inbound queue is in the hold status, the queue server task will immediately exit without processing any information.

If you are unsuccessful in sending messages between nodes with the Con-nect queues set to either "event" or "timer" status, use the following procedure to manually step through the process of sending and receiving a Distribution Interchange Unit (DIU):

- Initialize the node. A routing entry, an inbound queue, and one outbound queue must be defined to the node.
- Send a mail item from Con-nect. Then, check the inbound queue to see if the DIU is there.
- Start the inbound queue from the "Queue Maintenance" screen. This activates the router, which should route the DIU to the outbound queue. Since the outbound queue is inactive, the DIU should remain in the outbound queue.
- Check to see if the DIU was routed to the outbound queue.
- Check the log records. The log records can be accessed using the Log Information Maintenance function in the "SNADS Administration Menu" screen. A list of log records written by the SNADS driver programs are displayed. If the router was executed, log records indicating successful or unsuccessful routing are displayed. The exact format of these records is not available; however, you can verify whether or not tasks have been executed. Typically, a message, e.g. "routing error", which corresponds to a particular log record indicates that an error has occurred.

Up to this point, LU6.2 has not been involved. The following steps allow you to test sending a DIU to another SNADS node.

- Set the outbound queue with the routed DIU to "inactive" status; then start the outbound queue from the "Queue Maintenance" screen.

- Check the log records. The log records are accessed from the Log Information Maintenance function in the "SNADS Administration Menu" screen. If the DIU was successfully sent, records from the DS_SEND program (X-FON000) are displayed.
- Check for log records which were created when a status request was received from the target node. Records from the DS_RECEIVE program (X-FIN000) should be displayed.
- Check to see whether the DIU is delivered to the recipient node (the Inbasket, if Con-nect is the recipient node). In addition, there should be a "status" DIU generated and routed for delivery to the sender at the origin node.
- If you are still experiencing problems, the following commands and functions can be useful for debugging purposes:
 - CICS Supplied Transactions CEMT, CEDC and CEDF

For example, use the `CEMT I CONN` command to verify whether or not an LU6.2 session has been acquired (where "conn" is the connection ID). Use the `CEDC VIEW CONNECTION(conn) GROUP(isc)` command to examine the characteristics of an LU6.2 connection. Use the `CEDF` command to display the online trace utility.

- Con-nect SNADS Administration Functions

The following functions are offered in the Con-nect SNADS Administration Menu:

Queue Maintenance - Provides a DIU count, time stamps for recent activity, and the ability to manually start the queue server.

Control Maintenance - Provides the correlation between a DSUN and a system file (DBID and FNR).

Messages Awaiting Confirmation of Delivery - Displays the number of undelivered DIUs sent from Con-nect.

Log Information Maintenance - Provides debugging information that can be helpful to Software AG in resolving any problems you are experiencing with Con-nect SNADS.