

Installing NonStop SQL/MP

After ensuring that your node meets the hardware and software requirements for NonStop SQL/MP, you can install the NonStop SQL/MP relational database management system (RDBMS).

After installing NonStop SQL/MP, you can install the sample application distributed with the NonStop SQL/MP software, as explained in the *NonStop SQL/MP Reference Manual*. Use this application to demonstrate embedding SQL statements in host language programs, querying the catalogs, and querying and updating sample database tables. The sample database provided with this application is the same database used in examples in the NonStop SQL/MP manuals and education courses.

Note. If you are upgrading your NonStop SQL/MP environment from an older version of NonStop SQL/MP, consider the issues discussed in the *NonStop SQL/MP Version Management Guide* as well as those discussed in this section before running your existing NonStop SQL/MP applications.

This section describes system requirements for NonStop SQL/MP, installation procedures, migration to a newer version, and the use of D-series features (for users migrating from a C-series node).

Hardware and Software Requirements

The hardware and software requirements for a NonStop SQL/MP RDBMS are as follows:

- The hardware on which NonStop SQL/MP runs must be a NonStop SQL/MP system.
- Each NonStop system that includes NonStop SQL/MP requires a node (system) name, regardless of whether the node stands alone or is part of a network.
- The version of the operating system must be D30 or later to support versions 315 and newer of the NonStop SQL/MP software.
- As a general rule, the Transaction Management Facility (TMF) must be available when users are running NonStop SQL/MP application programs or using the conversational interface, SQLCI. The TMF subsystem is required for SQL compilation and for execution of all DDL statements, all SQL utilities, and DML statements that require TMF transactions for audited tables or views. TMF transactions are not required for previously compiled SQL statements that refer to nonaudited tables or views or for SELECT statements that use BROWSE ACCESS.
- The version of the TMF subsystem must be D30 or later to support versions 315 and newer of NonStop SQL/MP software.
- All SQL objects must reside on volumes audited by the TMF subsystem. SQL programs need not reside on audited volumes.
- If you plan to use a national character (NCHAR) data type from NonStop SQL/MP, the system default multibyte character set must be a character set supported by

NonStop SQL. To check the default character set, use the Guardian `MBCS_DEFAULTCHARSET_` procedure. To specify a different character set, rename the appropriate `LIBOBJnn` library object file to `LIBOBJ` before using `SYSGEN` to generate the node. `SYSGEN` accepts only one `LIBOBJ` file.

- For a list of supported character sets, see “Defining Columns” on page 5-19.

Recommendations for a node running NonStop SQL/MP are as follows:

- Mirrored volumes are recommended, but not required, for volumes containing SQL objects.
- A minimum of 16 megabytes of memory is suggested for each CPU in a node running NonStop SQL/MP. Additional memory can improve performance.

NonStop SQL/MP Software Components

The NonStop SQL/MP relational database management system consists of the following software components:

- BACKUP and RESTORE utilities
- Disk process
- FastSort software
- SQL catalog manager (SQLCAT)
- SQL compiler (SQLCOMP)
- SQL compiler interface
- SQL conversational interface (SQLCI)
- SQL executor
- SQL file system
- SQL utilities (SQLUTIL)
- Collation compiler
- Audit server (AUDSERV)

Installing NonStop SQL/MP

You should follow any instructions given in the associated software release document and any installation instructions that come with your site update tape (SUT). Install the software from the SUT by using the `INSTALL` program provided with the tape. The `INSTALL` program handles the installation of the NonStop SQL/MP system software on your `$SYSTEM` disk or alternate boot disk.

Starting the Transaction Management Facility (TMF)

Typically, you develop TMF startup and configuration files as OBEY command files. These files contain the TMF configuration options and parameters that describe the audit trails and the dump process. For information about TMF auditing requirements, configuration guidelines, and considerations for NonStop SQL/MP, see “The TMF Subsystem” on page 4-9 and the *NonStop TM/MP Configuration and Planning Guide*.

Your version of TMF must be compatible with your version of NonStop SQL/MP software as noted previously under “Hardware and Software Requirements” on page 2-1.

When you start the TMF subsystem, the configured data volumes are started for transaction processing if they are accessible. You must ensure the volumes you intend to use for NonStop SQL/MP catalogs and audited objects are configured and started for transaction processing. You must also ensure that the TMF subsystem as a whole is started for transaction processing. You can request status from TMFCOM, the TMF command interface, by entering the TMFCOM command at the operating-system command interpreter prompt:

```
21> TMFCOM

~STATUS TMF                                <--Request for TMF status.

TMF Status:
System: \SQLNLS, Time: 4-Nov-1994 13:35:26
State: started
Transaction Rate: 5.0 TPS
AuditTrail Status:
Master
Active audit trail capacity used: 14%
First pinned file: $DATA.ZTMFAT.AA000012
Reason: Current file
Current file: $DATA.ZTMFAT.AA000012
BeginTrans Status: ENABLED
Catalog Status:
Status: up

~STATUS DATAVOLS                          <--Request for status of data volumes.

~status datavols
  Audit   Recovery
Volume   Trail    Mode    State
-----
$RAT     MAT      Online  Started
$D00     MAT      Online  Started
$XCEED   MAT      Online  Started
$C30SYS  MAT      Online  Started
$SYSTEM  MAT      Online  Started
$SQL     MAT      Online  Started
$TES     MAT      Online  Started

~EXIT                                       <--Exits TMFCOM.
```

As an alternative to TMFCOM, you can use the TM/MP graphical user interface, TM View, to monitor and control the TMF environment. If you plan to use a system management program to operate TMF, you can use TMFSERVE, a TMF process that provides access to TMF by using the Subsystem Programmatic Interface (SPI). All three of these mechanisms can be used to monitor and control TMF operation.

For configuration information, see “Guidelines for Configuring TMF” on page 4-13.

Initializing NonStop SQL/MP

Before you can use NonStop SQL/MP for the first time, or when reinstalling the product, you must request initialization of NonStop SQL/MP by using the CREATE SYSTEM CATALOG and INITIALIZE SQL commands. The *NonStop SQL/MP Reference Manual* describes the syntax of these SQLCI commands.

SQL initialization involves creating the system catalog, SQL compiling the SQLCI2 program as a valid SQL program, then registering the program in the system catalog. The SQLCI2 program, when in execution, serves as a backend process for SQLCI.

The system catalog is like any other catalog, except it contains an additional table, CATALOGS, which is the system directory of catalogs. The CATALOGS table must reside on a subvolume named SQL.

By default, the system catalog resides on the subvolume \$SYSTEM.SQL; however, you can specify another volume and subvolume in the CREATE SYSTEM CATALOG command. If the system catalog is not on the subvolume SQL, the CATALOGS table is placed on the same volume as the system catalog, but on a subvolume named SQL.

For information about locating and securing catalogs, see “Creating Catalogs” in Section 5.

Steps for Performing Initialization

To initialize NonStop SQL/MP, follow these steps:

1. Check that the TMF subsystem is configured correctly and started.
2. Check that the \$SYSTEM.SYSTEM.ZZSQLCI2 file exists on the node and save a copy. This temporary file, which contains the SQLCI2 program, is copied onto the disk by the INSTALL program.

Note. As a basic rule, Tandem recommends you save a copy of the \$SYSTEM.SYSTEM.ZZSQLCI2 (SQLCI2) program at all times. After the SQL initialization is complete (Step 6), the temporary copy (ZZSQLCI2) is renamed and becomes the permanent SQLCI2 program. If a subsequent SQL initialization is attempted or if the SQLCI2 program is corrupted or purged, the saved copy provides a backup.

3. Logon as the local super ID. The super ID is required to execute the CREATE SYSTEM CATALOG and INITIALIZE SQL commands in Steps 5 and 6.
4. Start the SQLCI program. At the command interpreter prompt, enter the program name:

```
23> SQLCI
```

5. If you are installing SQL for the first time on this node, create the system catalog. If you are reinstalling it, you can skip this step.

By default, the system catalog resides on the subvolume `$SYSTEM.SQL`; however, you can specify an alternative volume and subvolume.

The system catalog contains a table called `CATALOGS`, which is the system directory of catalogs. If you put the system catalog on a volume other than `$SYSTEM`, NonStop SQL/MP puts the `CATALOGS` table on a subvolume named `SQL` on the same volume as the rest of the system catalog.

Note. It is recommended that you do not place the system catalog on the `$SYSTEM` volume. When the system catalog resides on another volume, the `$SYSTEM` volume can function as a nonaudited volume and can also be rebuilt from a system image tape (SIT), in case of disaster, without affecting the NonStop SQL/MP catalog structure.

If you want the system catalog to reside on the default location (`$SYSTEM.SQL`), enter this command at the SQLCI prompt:

```
>> CREATE SYSTEM CATALOG;
```

If you want the system catalog to reside on a volume or subvolume other than the default location, enter the following command at the SQLCI prompt:

```
>> CREATE SYSTEM CATALOG $vol.subvol;
```

In this command, *`$vol.subvol`* is a Guardian volume and subvolume name. If you do not specify a subvolume, the RDBMS uses the name `SQL` by default. If you specify a subvolume other than `SQL`, the RDBMS places all system catalog tables except `CATALOGS` on the subvolume you specify and places the `CATALOGS` table on a subvolume named `SQL`, on the same volume as the other catalog tables.

If you are installing NonStop SQL/MP on a system using the ServerWare Storage Management Foundation (SMF) product, and you want to ensure that you can fall back to a non-ServerWare SMF system, you should make sure that the system catalog tables reside on one physical volume. If you specify a virtual volume for the system catalog, ServerWare SMF can distribute the system catalog tables among multiple physical volumes in the storage pool. When this configuration is in place, there is no guarantee that you can return to using a nonvirtual volume. When you are certain you will not need to fall back to a non-ServerWare SMF system, you can specify a virtual volume for the system catalog tables without being concerned with the physical location of the files.

To ensure that the system catalog tables reside on one physical volume, you can specify a direct volume that is not in any storage pool, or you can use the `PHYSVOL` option, as follows:

```
>> CREATE SYSTEM CATALOG $virtual_vol.subvol
    PHYSVOL $physical_vol;
```

With the `PHYSVOL` option, you specify only the volume name. Also, the virtual volume specified with the `CREATE SYSTEM CATALOG` clause must be associated with the same storage pool that contains the physical volume specified with `PHYSVOL`. For more information about using this option, see the

NonStop SQL/MP Reference Manual and the ServerWare Storage Management Foundation User's Guide.

6. Initialize NonStop SQL/MP by entering this command at the SQLCI prompt:

```
>> INITIALIZE SQL;
```

During the installation of the NonStop SQL/MP system software, the INSTALL program places the new SQLCI2 program on the \$SYSTEM.SYSTEM subvolume in the file named ZZSQLCI2. SQLCI2 is the process through which the NonStop SQL/MP conversational interface (SQLCI) communicates with the NonStop SQL/MP executor to request various functions.

The SQL initialization process requested in this step drops the older version of SQLCI2 if it exists on the system, renames the ZZSQLCI2 file to SQLCI2, SQL compiles the program in \$SYSTEM.SYSTEM.SQLCI2, and registers the program in the PROGRAMS table of the system catalog. To execute SQL statements from SQLCI, SQLCI2 must be a valid, registered SQL program.

The INITIALIZE command performs the installation operations automatically. You can request these operations directly, however, by entering the following commands at the command interpreter prompt (for installation on \$SYSTEM.SYSTEM):

```
24> PURGE $SYSTEM.SYSTEM.SQLCI2    (if you are reinstalling SQL)
25> RENAME $SYSTEM.SYSTEM.ZZSQLCI2, $SYSTEM.SYSTEM.SQLCI2
26> SQLCOMP/IN SQLCI2/ CATALOG system-catalog
```

In the SQLCOMP command, *system-catalog* is the Guardian name of the system catalog.

7. Terminate SQLCI and create a backup copy of the collation compiler, which is in the \$SYSTEM.SYSTEM.NLCPCOMP file:

```
>> EXIT;

26> FUP DUP $SYSTEM.SYSTEM.NLCPCOMP, &
26> & $VOLBK.SUBVBK.NLCPCOBK, SOURCEDATE
```

(The collation compiler translates character-processing rules specified in a source file into an internal format.)

Then use the SQLCOMP command to SQL compile the collation compiler:

```
27> SQLCOMP /IN $SYSTEM.SYSTEM.NLCPCOMP,NOWAIT/ CATALOG
$VOL1.SQL
```

In this command, \$VOL1.SQL is the subvolume where the system catalog resides.

Note. You must SQL compile the collation compiler manually any time you install an IPM to the T6570 - National Language Character Processing product.

8. If you are installing SQL for the first time on this node, set the appropriate security string for the system catalog. You can use the ALTER CATALOG statement to alter the owner ID and security string of all the system catalog tables at once. Then, you

can use ALTER TABLE statements to resecure the CATALOGS, USAGES, TRANSIDS, and PROGRAMS tables separately.

```
>> ALTER CATALOG system-catalog SECURE "NG--";
```

```
>> ALTER TABLE system-catalog.CATALOGS SECURE "NA--";
```

For more information, see “Securing the System Catalog” on page 5-10.

9. Verify that the following programs are secured for execute access on your node:

```
$SYSTEM.SYSTEM.SQLCI
$SYSTEM.SYSTEM.SQLCI2
$SYSTEM.SYSTEM.SQLCAT
$SYSTEM.SYSTEM.AUDSERV
$SYSTEM.SYSTEM.SQLCOMP
$SYSTEM.SYSTEM.SQLESP
$SYSTEM.SYSTEM.SQLESPMG
$SYSTEM.SYSTEM.SQLUTIL
$SYSTEM.SYSTEM.NLCPCOMP
$SYSTEM.SYSnn.RECGEN
$SYSTEM.SYSnn.SORTPROG
```

In the preceding list, *nn* represents two digits assigned during node generation (SYSGEN operation).

Verify that the following file is secured for read access on your node:

```
$SYSTEM.SYSTEM.SQLMSG
```

If your node is part of a network, check that the programs in the preceding list are secured for network execute access and that the \$SYSTEM.SYSTEM.SQLMSG file is secured for network read access.

10. Verify that the following Guardian utilities and NonStop SQL/MP components are licensed so that they can access NonStop SQL/MP objects. The INSTALL process, when completed normally, performs the licensing of these programs (automatically or manually).

```
$SYSTEM.SYSTEM.SQLCOMP
$SYSTEM.SYSTEM.SQLUTIL
$SYSTEM.SYSTEM.SQLCAT
$SYSTEM.SYSTEM.AUDSERV
$SYSTEM.SYSnn.BACKUP
$SYSTEM.SYSnn.DSAP
$SYSTEM.SYSnn.FUP
$SYSTEM.SYSnn.FILCHECK
$SYSTEM.SYSnn.PUP
$SYSTEM.SYSnn.RESTORE
$SYSTEM.SYSnn.SORT
$SYSTEM.SYSnn.SORTPROG
```

In the preceding list, *nn* represents two digits assigned during system generation (SYSGEN operation).

You can use the FUP SECURE command to alter the security of these programs and the FUP LICENSE command to license the programs, if necessary. To alter the security of

the SQL sensitive program SQLCI2, you can use either the SQL ALTER PROGRAM statement or the FUP SECURE command.

Note. The SQLCI2 program is not licensed under normal circumstances. Only the super ID can license the SQLCI2 program. For more information, see Appendix A, “Licensed SQLCI2 Process.”

1. If old versions of SORT, SORTPROG, and RECGEN programs still exist on \$\$SYSTEM.SYSTEM, remove them now by entering the following command at the command interpreter prompt:

```
26> PURGE $$SYSTEM.SYSTEM.SORT, $$SYSTEM.SYSTEM.SORTPROG,
        $$SYSTEM.SYSTEM.RECGEN
```

If SORT, SORTPROG, or RECGEN programs exist on \$\$SYSTEM.SYSTEM from a previous release, the results are unpredictable.

NonStop SQL/MP uses the FastSort programs SORT and SORTPROG for sorting operations and the RECGEN program for parallel loading of indexes. The INSTALL process moves these programs to the SYSnn subvolume (in which nn is two digits assigned during system generation).

2. Verify that the SORT, SORTPROG, and RECGEN programs on the SYSnn subvolume are secured for execute access on your node. If your node runs in a network, check that these programs are secured for network execute access. If NonStop SQL/MP initiates a SORT operation and the SORT, SORTPROG, or RECGEN security does not allow access, a run-time sort error, Sort Start Error 4, is generated.

You can use the FUP SECURE command to alter the security of these programs, if necessary.

3. Check that the SQLMSG file you are using is the one released with the new version of NonStop SQL/MP. The message file contains error messages, warning messages, and help text for NonStop SQL/MP. You can check the version by running SQLCI and entering the ENV command; the message file version is listed as MESSAGEFILE VRSN in the ENV output.

Check that the following message files are secured for read access on your node:

```
$$SYSTEM.SYSTEM.SQLMSG
$$SYSTEM.SYSTEM.NLCPMSG
```

If your node is part of a network, check that these message files are secured for network read access. If necessary, secure them for network access by entering the following command:

```
32> FUP SECURE ($$SYSTEM.SYSTEM.SQLMSG, &
32> & $$SYSTEM.SYSTEM.NLCPMSG), "NN-N"
```

4. If you use COBOL85 with NonStop SQL/MP, replace the COBOLEXT file on your \$\$SYSTEM.SYSTEM subvolume with the correct COBOL85 extension file. The \$\$SYSTEM.SYSTEM.COBOLEXT extension file installed by the INSTALL program does not contain the complete COBOL85 extension libraries for

NonStop SQL/MP, which are stored under the name `$SYSTEM.SYSTEM.COBOLEX0`.

To replace the existing `$SYSTEM.SYSTEM.COBOLEXT` file with the appropriate COBOL85 extension file, enter this command:

```
27> FUP DUP $SYSTEM.SYSTEM.COBOLEX0, $SYSTEM.SYSTEM.COBOLEXT,  
PURGE, SOURCEDATE
```

If an incorrect extension file is used for programs containing SQL statements, those programs might encounter a compilation error (UNIT OF PROPER LANGUAGE NOT FOUND) or a run-time trap error.

Setting Up Event Logging

SQLCI provides logging capability to do the following:

- Log certain events to a terminal or file automatically
- Log command strings entered interactively through SQLCI, and, optionally, output from those commands, to an EDIT file

Both methods are effective for maintaining a record of events and commands. You should routinely have the log file duplicated, printed, and cleared. For more information about event logging, see the discussion of the `=_SQL_CMP_EVENT DEFINE` in the *NonStop SQL/MP Reference Manual*.

Setting Up Alternate SQL Components

Users with the super group ID can specify alternate SQL components through the `=_SQL_component` DEFINES. These DEFINES redirect the RDBMS to use the specified programs or message files instead of the default programs or files residing on `$SYSTEM.SYSTEM`. The use of these components is limited to members of the super group only.

If you want a NonStop SQL/MP system available only for the super group, issue the `=_SQL_component` DEFINES before performing the initialization or reinstallation steps. You might do this, for instance, when you want to allow a limited group of users to access a customized NonStop SQL/MP system or to run SQL components not residing on `$SYSTEM.SYSTEM`. For more information on the `=_SQL_component` DEFINES, see the *NonStop SQL/MP Reference Manual*.

Additional Installation Considerations

Installation of NonStop SQL/MP might require installation of a specific version of related software. For versioning requirements of products associated with NonStop SQL/MP, check the documentation supplied with your software.

If you install an IPM to the T6570 - National Language Character Processing product, you must SQL compile the collation compiler manually. For more information, see Step 7 under “Initializing NonStop SQL/MP” on page 2-4.

If you plan to define large numbers of partitions for tables and indexes in the database, consider using the `=_SQL_CAT_HEAP_LIMIT` DEFINE to increase the heap space

size limit for the catalog manager process. For more information, see the *NonStop SQL/MP Reference Manual* or SQLCI online help. You can specify this DEFINE in the TACLCSTM file for TACL sessions running applications or interactive queries.

Reinstalling NonStop SQL/MP Software

To reinstall NonStop SQL/MP on a node that has previously run the software, follow the steps listed under “Installing NonStop SQL/MP” on page 2-2.

Note that if you have installed NonStop SQL/MP on your node before, you do not need to re-create the system catalog.

Migrating to a Newer Software Version

To replace an existing version of the NonStop SQL/MP software, install the newer version of the software as described under “Installing NonStop SQL/MP” on page 2-2.

After you install the new version of NonStop SQL/MP software, follow these guidelines:

- Continue to run existing applications without recompilation.
- Continue to use existing local user catalogs until you are ready to start using the new features provided with the newer version of NonStop SQL/MP. You can continue to use your existing system catalog indefinitely, unless you plan to register newer-version objects in the system catalog.
- Continue to access all local and remote objects in your database, except for remote objects registered in catalogs whose versions are newer than the version of the NonStop SQL/MP software on either the local or remote node. For information about managing a network that has multiple versions of software, catalogs, and objects, refer to the *NonStop SQL/MP Version Management Guide*.
- Test the newly installed software before you use new features. This evaluation might require running the new software for several days of testing or simply running your applications under the new software for however long is appropriate. You can also test the new software by using the sample database distributed with your older release or by using your own test database.
- Test the features available with the newer version of software. Create test catalogs and objects for testing purposes, rather than altering existing objects. For example, collations require a version 300 or newer catalog in which to register the collation. Before adding a collation, create a test catalog to associate with the collation. Similarly, do not add a column with a new data type to an existing production table, because the table will not be accessible if you need to revert to an older software version.

If you need to revert to the older release, you can do it easily at this point because the versions of production catalogs and objects are unchanged, and there are no incompatible TMF audit records. (Guidelines for reverting are discussed under “Reverting to an Older Software Version” on page 2-15.)

Do not perform any of the following actions until you are reasonably sure you do not want to return to the older version of NonStop SQL/MP software:

- Create a production catalog whose version is newer than your previous version of NonStop SQL/MP software. A newer-version catalog is created automatically when you create a catalog on a node running the new version of the software.
- Create a production table using features that cause the version of the table to be newer than your previous version of NonStop SQL/MP software.
- Compile a program using the newer version of the NonStop SQL/MP compiler. If you do recompile a program and decide to return to an older version of software, you must recompile the program with the older version of the SQL compiler after reverting back to the older version. If you decide to compile programs with the newer version of software, keep track of which programs you compile.

When you are reasonably sure you do not want to revert to the older version of software, perform the following steps as needed based on your use of new features available in the newer version of software:

1. Upgrade catalogs. After you finish testing the newer version of the NonStop SQL/MP software, upgrade the user catalogs on your node as needed to use version 300 or newer features.

Note. You do not need to upgrade user catalogs until you are ready to register objects or programs with newer-version features. You can also continue to use your older-version system catalog indefinitely unless you register newer-version user objects or programs in the system catalog, because newer-version user catalogs can be registered in a older-version system catalog. For example, you can register version 300 catalogs in a version 1 or version 2 system catalog. You cannot, however, register version 300 or newer objects in a version 1 or version 2 system catalog.

If you are migrating from version 1 to a version 300 or newer version, you can use the following version 2 features without upgrading any version 1 catalogs: parallel execution, local autonomy, parallel index maintenance, and virtual sequential block buffering.

You can have a combination of versions of catalogs on a node. Unless you have an important reason for keeping this combination (such as operating in a mixed-version network), you should eventually upgrade all the catalogs on the node to simplify your database management efforts.

For additional information, see “Upgrading Catalogs” on page 2-13.

2. Recompile applications if desired. You do not need to SQL compile your application programs unless you want to take advantage of new features or performance enhancements.

To take advantage of new features, you might have to modify your programs. For example, versions 325 and newer software support CASE expressions. To use them, you need to modify source code and then host-compile and SQL-compile the modified programs by using the newer versions of the NonStop SQL/MP compilers. For more information, see the *NonStop SQL/MP Programming Manual* for the language you are using.

To take advantage of performance enhancements, host-compile and SQL-compile existing programs to ensure that the execution plan is optimal for the new

NonStop SQL/MP software. For example, versions 310 and newer SQL compilers generate query execution plans that provide better performance than plans generated by older-version SQL compilers.

With the exception of a dynamic SQL program that does not use the `RELEASE` option, a program written for an older version of the NonStop SQL/MP software should compile and execute on a node that has version 310 or newer software with no source code changes.

C-Series to D-Series Migration Considerations

If your previous installation of NonStop SQL/MP was on a node that ran a C-series release of the operating system, you should be aware of differences between the C series and D-series systems that affect the operation of NonStop SQL/MP.

The D-series features that might affect your NonStop SQL/MP applications are high and low process identification numbers (PINs), changes to file naming rules, and subvolume defaulting. In addition, you must understand the rules for combining C-series and D-series object modules.

For more information about D-series features, see the *D-Series System Migration Planning Guide*.

Mixed-Version Network Considerations

NonStop SQL/MP processes run at a high PIN by default. However, any process that needs to access objects on C-series nodes must not run at a high PIN. Therefore, the following NonStop SQL/MP processes must run at low PINs if they communicate with a process on a C-series node (or any process that cannot respond to a high-PIN process):

- FastSort (RECGEN, SORT, SORTGEN)
- SQL catalog manager (SQLCAT)
- SQL compiler (SQLCOMP)
- SQL conversational interface (SQLCI)
- SQL executor server processes (SQLESP)
- SQL utilities (SQLUTIL)
- SQLCI2
- Collation compiler (NLCPCOMP)

There are several approaches for running SQL programs at low PINs, including the following:

- Setting the `TACL HIGHPIN` environment variable to `OFF` when running any SQL program:
 - Arrange to have all TACL processes inherit a `HIGHPIN OFF` setting, either by specifying `SET HIGHPIN OFF` in the initial TACL process (before starting all other TACL processes on the node) or by placing a `SET HIGHPIN OFF`

command into the \$SYSTEM.SYSTEM TACLCSTM file, executed by every TACL process during logon.

- Alternatively, specify HIGHPIN OFF as needed in TACL RUN commands or SQLCI OBEY files.

Remove the HIGHPIN commands when access to C-series nodes is no longer needed.

- Setting the HIGHPIN attribute to OFF in all object files by using the BIND CHANGE command to change the HIGHPIN attribute for SQL program files after they have been installed. The syntax is:

```
BIND CHANGE HIGHPIN OFF IN $SYSTEM.SYSTEM.<filename>
```

where <filename> is SQLCI, SQLCI2, SQLCAT, SQLUTIL, SQLCOMP, SQLESP, RECGEN, and NLCPCOMP.

Before making this change, note the license flag setting for each file and relicense any licensed files after changing the HIGHPIN attribute. If you use this approach, continue to change the HIGHPIN setting after any subsequent installation of SQL software until access to C-series nodes is no longer needed.

For parallel index creation or parallel index load operations that access base table partitions on nodes running older versions of NonStop SQL/MP, specify the LOCALONLY option at the start of the CREATE INDEX configuration file to force RECGEN processes to run on the local node. Alternatively, start the CREATE INDEX operation from the node running the older version of software.

-
- △ **Caution.** If an SQL object has the UNRECLAIMED FREESPACE (F) or INCOMPLETE SQLDDL OPERATION (D) attribute set, do not attempt to back up, move, or duplicate the object until the attribute is reset. For more information, see “UNRECLAIMED FREESPACE (F) and INCOMPLETE SQLDDL OPERATION (D) Flags” on page 7-22.
-

Upgrading Catalogs

You must upgrade or create at least one user catalog on a node running the new version of NonStop SQL/MP software before you can do any of the following:

- Create an object with attributes that cause the object version to be newer than the current version of the user catalog. For example, you cannot create an object that references a version 320 feature and register the object in a version 1 or version 2 catalog.
- Alter an object by adding attributes that cause the object version to be newer than the current version of the user catalog. For example, you cannot add a column with a version 310 data type to a table registered in a version 1 or version 2 catalog.
- Register programs with a program catalog version newer than the current version of the user catalog. For example, you cannot register a program that contains a version 320 feature in a version 1 or version 2 catalog.

To create a version 300 or newer catalog, use the CREATE CATALOG command.

To upgrade an existing user catalog to version 300 or newer, use the UPGRADE CATALOG command. This command requires exclusive access to the catalogs being upgraded and to the objects to be registered in the catalogs. Specifically, to use UPGRADE CATALOG, you must be the local owner of the catalog, the local super ID, the group manager, or the remote owner with authority to purge the catalog tables. You must also have authority to write to the CATALOGS system catalog table.

NonStop SQL/MP automatically requests the TMF subsystem to protect the integrity of the database during the upgrade operation. If a user-defined TMF transaction (a transaction explicitly defined by using language statements such as BEGIN WORK and COMMIT WORK) is not in progress when you enter the UPGRADE CATALOG command, NonStop SQL/MP begins and ends a TMF transaction for each catalog being upgraded.

For statements issued within a user-defined TMF transaction, SQLCI does not initiate a system-defined TMF transaction. You should allow SQLCI to initiate TMF transactions for DDL commands.

You can use a single UPGRADE SYSTEM CATALOG command to upgrade all the catalogs on a node; however, doing so can take several minutes to complete and, during this time, no user or program can access any catalogs or objects on the node. For this reason, you might want to upgrade catalogs individually.

The following example upgrades three user catalogs, one catalog at a time:

```
>> UPGRADE CATALOG \SYS1.$VOL1.SALES TO 310;
--- SQL operation complete.
>> UPGRADE CATALOG \SYS1.$VOL1.INVENT TO 310;
--- SQL operation complete.
>> UPGRADE CATALOG \SYS1.$VOL1.PERSNL TO 310;
--- SQL operation complete.
```

You can continue using the existing system catalog and create versions of catalogs that are newer than the current system catalog (but not newer than the NonStop SQL/MP software version on the node on which the catalog resides). For example, you can create a version 310 catalog to be registered in a version 1 or version 2 system catalog residing on a version 310 NonStop SQL/MP node.

If you decide to register objects and programs in the system catalog, instead of in user catalogs, you must upgrade the system catalog before registering objects or SQL programs that have a newer version than the system catalog. To do this, use the UPGRADE SYSTEM CATALOG command, identifying the system catalog as the one to be upgraded.

Note. Due to database administration overhead, you should not register user objects in the system catalog. Also, the impact of upgrading a system catalog in a mixed-node environment can make this system inaccessible to other nodes in the network.

For the syntax and other examples of UPGRADE CATALOG and UPGRADE SYSTEM CATALOG, see the *NonStop SQL/MP Reference Manual*. For more information about catalog versions, see the *NonStop SQL/MP Version Management Guide*.

Reverting to an Older Software Version

If you have upgraded the system catalog to a newer version and you want to revert to an older version of NonStop SQL/MP, you must first drop all objects whose versions are newer than the version of NonStop SQL/MP to be reinstalled. You must then downgrade the system catalog and any user catalog with a version newer than the version of NonStop SQL/MP to which you are reverting. After you reinstall the older version of NonStop SQL/MP software, you must recompile newer-version programs.

Dropping Newer-Version Objects

To preserve data in tables that have a newer version than the software to which you are reverting, you must create each of the tables again, omitting any features that caused the table to have the newer version. For example, if tables use extended partition arrays, re-create them with standard partition arrays. If the tables you want to preserve are defined with new features, write a program to move the required data from the newer-version tables to the older-version tables or use SQLCI for this purpose. You must perform these tasks before installing the older-version software.

-
- △ **Caution.** If any tables have the UNRECLAIMED FREESPACE (F) or INCOMPLETE SQLDDL OPERATION (D) attribute set, remedy the situation before downgrading the version. Otherwise, the table might be corrupt after the downgrade has completed. For more information, see “UNRECLAIMED FREESPACE (F) and INCOMPLETE SQLDDL OPERATION (D) Flags” on page 7-22.
-

After preserving any necessary data, you can drop all newer-version objects. You can also drop any programs that must be registered in a newer-version catalog; this includes any program with a newer-version feature, which causes the program catalog version to adopt the newer version.

Downgrading Catalogs

If you need to access objects in a newer-version user catalog, you must downgrade the user catalog before installing the older version of NonStop SQL/MP software.

You must always downgrade a version 300 or newer system catalog when you revert to an older version of the software, but you do not always need to downgrade version 300 user catalogs. For example, if you are temporarily reverting to version 2 from version 310 and do not intend to access objects registered in user catalogs, you can leave any version 310 user catalogs and objects on the node and downgrade only the system catalog.

Note. You do not need to downgrade any version 2 user or system catalogs, because version 2 catalogs are compatible with version 1 NonStop SQL/MP software.

If, however, you need to access data from the version 310 user catalogs on the node where you are reinstalling an older version of NonStop SQL/MP, you must downgrade these catalogs. If you do not downgrade the catalogs on a node in which the NonStop SQL/MP software reverts to an older version, a fallback situation results. In this situation, the reinstalled NonStop SQL/MP software cannot access any catalogs whose versions are newer than the version of the software or any objects or programs

registered in the catalogs. In addition, NonStop SQL/MP software on other nodes in a network can no longer access these catalogs and objects, regardless of the versions of the software on the remote nodes.

- △ **Caution.** If you want to downgrade any catalogs, you must do so before you reinstall the older version of the NonStop SQL/MP software, because only version 300 or newer software supports the DOWNGRADE CATALOG and DOWNGRADE SYSTEM CATALOG commands.
-

Downgrading User Catalogs

Before downgrading a user catalog, you must do the following:

- Drop any protection views on the catalog tables to be downgraded.
- Drop any object whose version would be newer than the version of the catalog after the catalog is downgraded.
- Drop any program whose program catalog version would be newer than the version of the catalog after the catalog is downgraded. For example, drop version 310 and newer programs before reverting to version 2.

You cannot downgrade a catalog to version 1, but version 2 catalogs are compatible with version 1 NonStop SQL/MP software if you access only version 1 objects.

For information about catalog, object, and program versions, see the *NonStop SQL/MP Version Management Guide*.

You must have exclusive access to the catalog tables. You can downgrade more than one catalog by specifying a catalog name pattern. For a detailed description of the DOWNGRADE CATALOG command and the requirements for using it, refer to the *NonStop SQL/MP Reference Manual*.

Downgrading the System Catalog

Before you can downgrade the system catalog to a version older than 300, you must drop any version 300 or newer objects registered in the system catalog. You must also be logged on with the super ID.

You cannot downgrade the system catalog to version 1, but version 2 system catalogs are compatible with version 1 NonStop SQL/MP software.

For a description of the syntax and more examples of using the DOWNGRADE SYSTEM CATALOG command, refer to the *NonStop SQL/MP Reference Manual*.

Recompiling Programs

If you want to use a program compiled with a version 300 or newer SQL compiler (a program whose program format version is 300 or newer), you must recompile the program with the older-version SQL host compiler compatible with the version of NonStop SQL/MP you have reinstalled, then SQL compile the program with the appropriate SQL compiler.

Reverting to NonStop SQL/MP Version 2

To revert to version 2 of NonStop SQL/MP, follow these steps:

1. If you have any version 300 or newer tables you want to preserve, use the CREATE TABLE statement to create version 2 tables to contain any version 2 data you want to preserve, and use the LOAD utility to move the data from the version 300 tables to the version 2 tables.
2. Use the DROP statement to drop version 300 or newer tables, indexes, views, and collations.
3. Make a list of all programs whose program format version or program catalog version is 300 or newer. You can generate a list by entering a SELECT statement like the following for each catalog on the node:

```
>> SELECT PROGRAMNAME, PROGRAMFORMATVERSION,
+>   PROGRAMCATALOGVERSION
+> FROM $CATVOL.SYS.PROGRAMS    <--PROGRAMS table in system catalog
+> WHERE PROGRAMCATALOGVERSION >= 300
+> OR PROGRAMFORMATVERSION >= 300 ;
--- SQL operation complete.
```

To inquire about the version of a specific program, use the GET VERSION OF PROGRAM statement.

4. Use the DROP PROGRAM statement (or the corresponding OSS utility for OSS programs) to drop all programs whose program catalog version is 300 or newer.
5. Use the DOWNGRADE CATALOG command to downgrade all version 300 or newer user catalogs to version 2:


```
>> DOWNGRADE CATALOG catalog-name TO 2 ;
--- SQL operation complete.
```
6. If the system catalog is version 300 or newer, use the DOWNGRADE SYSTEM CATALOG command to downgrade the system catalog to version 2:


```
>> DOWNGRADE SYSTEM CATALOG TO 2 ;
--- SQL operation complete.
```
7. Reinstall version 2 of the NonStop SQL/MP software as described under “Reinstalling NonStop SQL/MP Software” on page 2-10.
8. If any programs you want to use were host compiled by a version 300 or newer host compiler, use a version 2 host compiler to recompile the source code for these programs.
9. If any programs you want to use have a program format version of 300 or newer (as determined in Step 3), use a version 2 SQL compiler to recompile these programs.

Reverting to an Older Version of TMF

When moving from version 300 or newer NonStop SQL/MP software back to an older version of software, you must also revert back to the older version of TMF software. For more information, see the *NonStop TM/MP Configuration and Planning Guide*.

