STORAGETEK®

PART NUMBER
312564001

VERSION NUMBER
1.1

EDITION NUMBER
1

LIFECYCLE DIRECTOR™

DB2 MANAGER USER GUIDE

PRODUCT TYPE
SOFTWARE

Lifecycle Director™ DB2 Manager User Guide

Version 1.1

EC 132056

Revision C

Information contained in this publication is subject to change without notice.

We welcome your feedback. Please contact the Global Learning Solutions Feedback System at:

GLSFS@Stortek.com

or

Global Learning Solutions
Storage Technology Corporation
One StorageTek Drive
Louisville, CO 80028-3256
USA

Please include the publication name, part number, and edition number in your correspondence if they are available.

Export Destination Control Statement

These commodities, technology or software were exported from the United States in accordance with the Export Administration Regulations. Diversion contrary to U.S. law is prohibited.

Restricted Rights

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) and (2) of the Commercial Computer Software - Restricted Rights at FAR 52.227-19 (June 1987), as applicable.

Limitations on Warranties and Liability

Storage Technology Corporation cannot accept any responsibility for your use of the information in this document or for your use in any associated software program. You are responsible for backing up your data. You should be careful to ensure that your use of the information complies with all applicable laws, rules, and regulations of the jurisdictions in which it is used.

Proprietary Information Statement

The information in this document, including any associated software program, may not be reproduced, disclosed or distributed in any manner without the written consent of Storage Technology Corporation.

Should this publication be found, please return it to StorageTek, One StorageTek Drive, Louisville, CO 80028-5214, USA. Postage is guaranteed.

First Edition (September 2004)

This edition contains189 pages.

Table of contents

Introduction

This Lifecycle Director DB2 Manager User Guide provides all the information required for installation, implementation and operation of the DB2 Manager product for the enabling of Lifecycle Director support for the storage and retrieval of table rows using IBM's DB2 relational database management software.

For ease of reference, the product will be referred to as DB2 Manager throughout the remainder of this manual. In addition, the Archive Manager component of Lifecycle Director which is used by DB2 Manager for storage of migrated data will be referred to as Archive Manager throughout the remainder of the document.

The manual assumes some familiarity with DB2 concepts and its implementation. Information on these topics may be obtained from the appropriate IBM manuals where necessary.

Version 1.1 of the product will support all releases of DB2 from version 4 upwards. Page 16 of the manual lists all the pre-requisite software and hardware requirements for DB2 Manager implementation.

Implementation of version 2.5 or higher of the Archive Manager tape database management component is a pre-requisite for DB2 Manager installation. Users should refer to the Archive Manager User Manual for the version of Archive Manager installed at their installation for information regarding that product component.

Any additional installation or release-dependent information not contained in this manual will be supplied to customers as part of the DB2 Manager distribution package.

Product Description

1

Introduction

Storage Technology's DB2 Manager software product is designed to implement Archive Manager support for storage of table rows using IBM's DB2 relational database management product. Archive Manager is a Storage Technology archival database management product, which primarily uses tape cartridge media for storage of archived objects. Archive Manager also optionally enables disk copies of objects in an Archive Manager database to be retained. The product supplies a range of facilities to optimize the storage and retrieval of objects in an archive database.

Installation of Archive Manager on the host system is a pre-requisite for DB2 Manager implementation. Migrated rows are stored as objects in a standard Archive Manager database, each Archive Manager database consisting of a discrete set of tape cartridge volumes (plus optional disk copy datasets). DB2 Manager uses a separate Archive Manager database for each DB2 table which has been enabled for archival with DB2 Manager.

DB2 Manager enables applications developed to use DB2 as a data manager to extend the range of storage options supported by DB2 to include an Archive Manager database. Version 1.1 of DB2 Manager includes support for all versions of DB2 from version 4 upwards.

Implementation of the product requires no modifications to applications which use standard SQL processing to access DB2 tables. This means that customer-developed or vendor-supplied DB2 SQL applications will be able to store and retrieve table rows in Archive Manager without modification.

As is the case for all Archive Manager applications, usage of DB2 Manager to access tape-resident objects in an online processing environment requires the implementation of an automated tape processing facility, using the StorageTek 4400 ACS range of products.

There are no functional limitations in DB2 Manager which would prevent its implementation in a manual tape-processing environment, using free-standing tape cartridge drives. However, the need for manual operator intervention in this environment would mean that SQL commands issued from an online processing system, which required access to a tape-resident object, would wait indefinitely for a response (depending on the time taken to manually satisfy the tape-handling request). A guaranteed level of service for processing these requests can only be supplied through the implementation of an automated tape handling strategy.

Note that a single SQL command issued from an application problem may generate access to multiple rows from one or more tables. In these instances, DB2 will create a temporary table containing the result of the SQL request. Standard DB2 processing then requires the use of a cursor mechanism in order to access each row from the result table, rows being retrieved via an SQL FETCH command. In cases where rows in the result table have been migrated from DB2 to Archive Manager, the resulting FETCH commands may generate one or more tape mounts in order to service the application program's SQL request. Access to rows is not necessarily performed in the physical sequence in which migrated rows are stored in the Archive Manager database – this may generate an excessive amount of tape mount activity, with a correspondingly negative effect on application response times.

For this reason, implementation of DB2 Manager for migration of data from a DB2 table may not be suitable in instances where applications process the table in the following manner:

- Sequential processing of the table. This can occur if the application needs to access all rows in the table (e.g. for historical data analysis), or if access to specific rows in the table is not performed via index processing.
- Many rows are required to satisfy a single SQL command. If migrated rows are among those required to satisfy the SQL command, then extended response times may be encountered.

Note that use of Archive Manager disk copy processing (where copies of objects in the Archive Manager database are held in sequential disk datasets) is enabled, then the above restrictions will not necessarily apply.

Only one DB2 Manager control region is started per MVS system – this will support access to migrated rows from multiple DB2 subsystems executing on that MVS system.

■ Product Functions

The functions supplied by the DB2 Manager product can be categorized into three main areas:

1. Migration of rows from a DB2 table.
2. Enabling of access to migrated rows.
3. Management of migrated rows.

These functions are discussed in the following sections.

Row migration

The primary function of DB2 Manager is to enable individual rows from within a DB2 table to be migrated from DB2 to an Archive Manager database. Migrated rows will be stored as part of a standard Archive Manager object in the Archive Manager database, and will be replaced in the VSAM tablespace dataset by an 18-byte archive "stub" which will contain information about the location of the row in the Archive Manager database. This will allow space within the table to be released and re-used, enabling a reduction in the overall size of the DB2 tablespace used for table storage.

Reduction in the tablespace size will give a corresponding reduction in the overall amount of disk space required to support DB2 database usage, and also reduce the amount of DB2 housekeeping processing (in particular, image-copy and re-organization processing) required for the database containing the table(s) enabled for DB2 Manager migration processing.

Migration of rows is performed using a batch utility which is supplied with the product (see page 80). Rows are selected for migration via customer-specified rules, which are specified using SQL command syntax, allowing maximum flexibility for selection of rows to be migrated.

In order to be eligible for migration of rows by DB2 Manager, a table must have the following characteristics:

- A DB2 edit procedure (OTDBP300) must be defined for the table. This edit procedure (known as the “DB2 Manager SQL intercept”) intercepts all row insert, update and retrieve operations performed by DB2 on the table, and determines whether migration of an unmigrated row is to be performed, or whether retrieval of a migrated row is required.
- The table must have at least two partitions. One of these partitions is used as the “archive partition”, and will be used exclusively for storage of the archive stubs for migrated rows. All unmigrated rows will be held in the one or more remaining partitions of the table.
- An additional single-character column must be defined for the table. This will contain an archive indicator identifying whether a row is migrated or unmigrated. Views of the table are defined to allow existing applications to access the table without modification.

Refer to the discussion on product implementation in chapter 2 of this manual for a full description of the actions required in order to enable a table for migration processing by DB2 Manager.

Access to migrated rows

DB2 Manager allows access to migrated rows without requiring any modifications to applications which use SQL to access tables containing migrated rows. All types of applications (batch, TSO, CICS, IMS etc.) are supported. Multiple migrated rows from a single table may be accessed via a single SQL command, using a cursor operation to retrieve individual rows. An application may also access migrated rows from more than one table via a single SQL command, using standard DB2 JOIN or UNION processing.

The DB2 Manager “control region” (an OS/390 started task) must be started in order to enable this retrieval support. All Archive Manager database access will be performed from the control region, allowing full shared access to the Archive Manager database(s) containing migrated data.

Retrieval of migrated rows

A migrated row which has been retrieved from an Archive Manager database is presented back to the calling application on reference via SQL. No indication is given to the application that the row has been retrieved from Archive Manager, and the contents of the row will be identical to those prior to migration of the row from DB2.

The following factors will govern the length of time taken to access a migrated row stored in a tape-resident object in an Archive Manager database:

- The time taken to mount a tape. This is dependent on the automatic library robot accessor utilization rate.
- The time taken to locate an object on a tape once mounted. This will depend on the amount of data held per Archive Manager tape volume (user-controlled via Archive Manager database initialization parameters).
- Tape drive availability. Lack of tape drives will prevent immediate recall of objects from tape.

DB2 Manager provides the following controls to optimize tape recall performance:

- Specification of the maximum number of tape drives to be used by DB2 Manager for object retrieval. These controls are similar to those provided by the online recall component of Archive Manager (ie) drives required by DB2 Manager for tape object retrieval will be dynamically allocated and released as required, up to the maximum specified in the DB2 Manager drive control parameters.
- Parameterized controls are provided to allow users to limit the length of queues of retrieval requests for any one specific tape volume. All requests subsequent to the first for a mounted tape will be satisfied by repositioning the tape. However, it may take up to 30 seconds to reposition a tape; in order to provide a control over the length of time taken to retrieve any one object, a limit may be placed on the number of requests queued for an active tape.

If insufficient resources are available to satisfy an object retrieval request, DB2 Manager will reject the request with the appropriate SQL error and reason codes, or may optionally queue the request internally until sufficient resources become available or until a customer-specified time interval has elapsed.

The above controls are specified using the DB2 Manager parameter library, and are processed during control region initialization. They may be dynamically adjusted during DB2 Manager operation via control region command processing.

Update of migrated rows

If an application updates a migrated row after retrieval from an Archive Manager database, the migrated copy of the row will be invalidated and the updated row will be stored back in the DB2 table from which it was originally migrated. All reference to the migrated copy of the row in the Archive Manager database will have been deleted, causing this row to be unreferenced. This will effectively “re-migrate” the row from Archive Manager to DB2. The updated row will be stored back in a non-archive partition of the table - the corresponding 18-byte archive stub in the archive partition will be deleted allowing this space to be re-used for another archive stub.

A migrated row which has been updated and stored back in DB2 will subsequently become eligible for re-migration to Archive Manager under the control of the DB2 Manager batch migration utility. Selection of that row via the SQL command used to control row migration for a table will cause it to be migrated from DB2 to Archive Manager – however, it should be noted that the migrated row will be stored in a different Archive Manager object from that used for storing the previous migrated copy of the row.

The Archive Manager object containing a migrated row which has been updated and restored in the DB2 table will continue to remain in the Archive Manager database until all migrated rows stored in that object have been invalidated through update or deletion processing. The DB2 Manager database housekeeping utility is used to delete Archive Manager objects which no longer contain any active migrated rows. Following deletion of the Archive Manager object, the base Archive Manager object management and database maintenance utilities may be used to reclaim tape space occupied by the deleted object, if required. Refer to chapter 6 of this manual for a detailed description of this process.

It should be noted that frequent updating of a row after it first becomes eligible for migration is likely to cause multiple instances of migration and re-migration activities, which may result in a high proportion of invalidated space in the Archive Manager database. Tables which are accessed by applications in this manner may not be appropriate for migration processing using DB2 Manager. Refer to page 14 for further discussion of this issue.

Deletion of migrated rows

Migrated rows may be deleted by an application program using an SQL DELETE command. Deletion of the row will cause the migrated copy of row to be retrieved from the Archive Manager database, and the archive stub will then be deleted from the archive partition of the DB2 table, causing all reference to the migrated row to be removed.

DB2 Manager housekeeping processing is used to synchronize deletion of a migrated row from the DB2 table with deletion of the Archive Manager object containing the migrated row, in an identical manner to that described for row update processing on page 11.

Management of migrated rows

A set of batch utilities is supplied with the product in order to assist in management of migrated rows. The following utilities are supplied with v1.1 of the product:

- Row restore utility. This will allow migrated rows to be returned to the table from which they were migrated. Customer-specified rules, using SQL command syntax, are used to select migrated rows for restore.
- Table analysis utility. This will analyze a table which has been enabled for row migration, and produce a detailed or summary report on the contents of the table. A detailed report will identify each row which has been migrated (by DB2 key), and identify the Archive Manager object containing the migrated row.
- Archive Manager database housekeeping utility. This will process a table which has been enabled for row migration, and delete any objects in the Archive Manager database associated with the DB2 table, which no longer contain any active migrated rows.

Detailed descriptions of each of the above utilities may be found in chapter 6 of this manual.

■ DB2 Manager storage configuration

Archive Manager is used for storage and retrieval of migrated rows, so all the functions and benefits of the base Archive Manager product are available for use with DB2 Manager.

One or more Archive Manager databases must be defined on the system for storage of migrated rows. One Archive Manager database is required for each DB2 table which has been enabled for DB2 Manager migration processing. DB2 Manager will enforce the one-for-one relationship between DB2 table and Archive Manager database during migration processing.

The Archive Manager database must be defined with a primary keylength of 4 bytes. Each object in the database will contain one or more migrated rows – the maximum number of migrated rows per object is controlled via the OBJSIZE parameter in the DB2 Manager ENVCNTL parameter library member. A migrated row will be stored as a single record within the object – the product does not currently include support for DB2 LOBs, so the maximum size of any row which is eligible for migration is 32k bytes.

The location of a migrated row within an Archive Manager database is determined by the following entities:

1. Archive Manager database identifier. This is a unique 4-character identifier established by the customer, and is used to identify a single Archive Manager database throughout the product. The identifier of the database to be used to hold a migrated row is specified via execution parameter to the migration utility.

2. Primary key. This is a 4-byte (fullword) binary value, which is automatically assigned by DB2 Manager during migration processing. A primary key value of 1 will be assigned to the object containing the first row archived from a table after it has been enabled for archival. All rows within a single Archive Manager object will have the same primary key identifier. DB2 Manager will increment the primary key by 1 when creating a new object.
3. Archive date. This is automatically assigned by DB2 Manager during archival processing, and will be equal to the system date on which the row is archived.
4. Record offset. This is the record number within the object, which contains the migrated row (starting at 0, and continuing up to n-1, where ‘n’ is the value of the OBJSIZE parameter at time of archival).

A concatenation of these four values will uniquely index each migrated row within the Archive Manager database. This information is held in the archive stub used by DB2 Manager to replace the migrated row in the DB2 table.

All facilities supplied by the base Archive Manager product for managing data within the Archive Manager database are available for use with an DB2 Manager implementation. This includes the following:

- Multiple tape copies
- Disk (‘K’) copy processing
- Disaster recovery support
- Multiple storage levels per database
- Tape recycling
- Tape backup/recovery support
- Dynamic load balancing

■ Product implementation

Implementation of DB2 archival solutions using DB2 Manager is aimed at tables and applications with the following characteristics:

- A high proportion of row retrieval requests are satisfied from a small proportion of the rows (eg) 90% of the requests reference only 10% of the rows. This can occur with tables which hold historical information – customer transactions etc.

  Data access patterns of this nature will permit an aggressive row archive policy to be implemented while limiting the impact on overall DB2 response times, giving significant benefits in terms of reduced disk space utilization and/or DB2 housekeeping overhead.
- Once archived, data rows are unlikely to be updated frequently. Updating of an archived row via SQL command will cause the row to be returned from Archive Manager storage to DB2 storage (ie) “unarchiving” the row. The row will be subsequently re-archived under control of the archival selection criteria specified by the customer.

Rows which have a high likelihood of being updated after archival may cause a significant amount of archival and re-archival processing. A table containing rows which are likely to be updated in this manner are not suitable for migration using DB2 Manager.

DB2 Manager does not impose limitations on the way in which rows are referenced from a DB2 application program. Every possible means of accessing a migrated row via SQL commands is supported, including joins and unions.

Multiple migrated and un-migrated rows from one or more tables may be accessed via a single SQL command from a DB2 application program. However, if a large number of migrated rows is referenced by execution of such a command, then it may take a substantial length of time for execution of that command to be completed if migrated rows are not resident on Archive Manager disk storage. Table migration via DB2 Manager may not be appropriate if retrieval requests of this nature are likely to be performed on a frequent basis.

Alternatively, in these circumstances replacement of a single SQL command via a row-by-row operation using DB2 cursor processing will give a program control over the number of retrievals of migrated rows which may be required to complete an operation – in this case there will be a one-for-one relationship between a single FETCH command and a single migrated row retrieval operation.

■ DB2 management processing

Database backup and recovery

There is no change required to backup and recovery processing procedures for tablespaces containing tables which have been enabled for DB2 Manager migration. Image copies should be taken normally as for non-enabled tables. However, as the size of the tablespaces used for storage of tables may now have been substantially reduced, the length of time taken for image-copy processing will be reduced by a corresponding amount.

Restore of a tablespace containing a migration-enabled table is performed in an identical manner to that of a tablespace containing a non-enabled table. DB2 will restore the 18-byte stubs containing information about migrated rows, as well as restoring non-migrated rows, to the condition they were in at the time the image-copy was taken. The stubs contain all the information necessary to access migrated rows from Archive Manager.

Database reorganization

Tables which have been enabled for migration processing must be defined in a partitioned tablespace. This tablespace will have one partition for exclusive storage of migrated row stubs, and one or more partitions for storage of non-migrated rows.

Database reorganization processing should **not** be performed on the partition containing the migrated row stubs, unless absolutely necessary. Reorganization of this partition will cause all migrated rows to be retrieved from Archive Manager. If Archive Manager objects are held on tape storage only, this is likely to generate a substantial amount of tape activity and could take an excessive length of time to complete. However, all rows stored in this partition will be a fixed 18 bytes in length. In this case, deletion of rows is not likely to cause fragmentation in the tablespace, as deleted space can be re-used as required. The requirement for reorganization of this partition is thus not mandatory.

Reorganization of other (non-migrated) partitions in the tablespace should be performed as normal. Regular reorganization will, in particular, be required for a period of time after implementation of DB2 Manager migration, in order to reclaim disk space occupied by rows which have been migrated to Archive Manager. This will allow these tablespace partitions to be reduced in size, eventually down to the size needed for the “working-set” of unmigrated rows (ie) rows which have yet to be migrated to Archive Manager.

■ Pre-requisites for DB2 Manager implementation

In order to use DB2 Manager version 1.1 on a system for implementation of Archive Manager support for storage and retrieval of DB2 table rows, the following software pre-requisites are required:

- OS/390 V1.1 or higher
- DB2 V4 or higher
- Archive Manager V2.5 or higher
- Support for 3480/3490/3590 device types.

The following hardware pre-requisites will be required:

- 3480/3490/3590-compatible cartridge tape devices.
- A correctly sized STK 4400 ACS library configuration.

Sizing of the hardware requirements is dependent on a number of factors relating to the volume and usage of DB2 tables which are to be eligible for row migration. This may be a complex task, and should be performed in conjunction with the StorageTek customer support representative for your installation.

This page is intentionally left blank

Installation and Implementation

2

This section describes the activities required to install and implement DB2 Manager for use in migrating and retrieving DB2 table rows to and from Archive Manager in a host system environment.

After all activities described in this section have been successfully completed, DB2 Manager will be fully installed and Archive Manager support for DB2 may then be enabled.

Archive Manager implementation

DB2 Manager uses the Archive Manager component of Storage Technology's Lifecycle Director product to perform its row migration and retrieval operations. Installation of Archive Manager on the same host system is therefore a mandatory pre-requisite for DB2 Manager implementation and use.

Prior to proceeding with DB2 Manager implementation, customers should verify that installation and implementation of Archive Manager has been completed successfully. Refer to the Archive Manager User Manual for details on implementation of this product. The installation verification procedure (IVP) supplied with the product should be executed to verify that Archive Manager has been successfully installed on the system on which DB2 Manager implementation is to be performed.

Note that DB2 Manager does not use CICS support within Archive Manager. Archive Manager implementation and IVP activities relating to CICS need not be undertaken unless this support is needed for other (non-DB2) applications within the organization.

DB2 Manager implementation procedures

The following activities are required when installing DB2 Manager. A full description of each activity is provided in the section indicated in brackets.

1. Install distribution libraries with SMP/E
2. Update DB2 Manager parameter library
3. Perform MVS host system modifications
4. Perform DB2 system modifications
5. Define Archive Manager databases

Install distribution libraries

DB2 Manager V1.1 is distributed on a standard-labeled magnetic tape cartridge in SMP/E RELFILE format. The cartridge has a serial number of OTD110. The distribution tape will contain the following datasets:

	Dataset Name	Description
1	SMPMCS	SMP/E modification control statements for installation of the product.
2	STK.SOTD110.F1	SMP/E JCLIN file.
3	STK.SOTD110.F2	Partitioned dataset in IEBCOPY UNLOAD format containing JCL for use during product installation.
4	STK.SOTD110.F3	DB2 Manager distribution load library, in SMP/E RELFILE format.
5	STK.SOTD110.F4	DB2 Manager distribution LPA library, in SMP/E RELFILE format.
6	STK.SOTD110.F5	DB2 Manager distribution DBRM library, in SMP/E RELFILE format.
7	STK.SOTD110.F6	DB2 Manager sample JCL library, in SMP/E RELFILE format.
8	STK.SOTD110.F7	DB2 Manager sample parameter library, in SMP/E RELFILE format.

To install DB2 Manager, perform the following steps:

1) Copy the installation JCL (file 3) from the distribution tape to disk. The following JCL may be used for this purpose:

```
//S1       EXEC PGM=IEBCOPY
//SYSPRINT   DD SYSOUT=*
//SYSIN      DD DUMMY
//SYSUT1     DD DISP=(OLD,KEEP),UNIT=unit-name,
//            VOL=(,RETAIN,SER=OTD110),
//            LABEL=(3,SL,EXPDT=98000),
//            DSN=STK.SOTD110.F2
//SYSUT2     DD DISP=(NEW,CATLG),
//            DSN=DB2 Manager.install.JCL.library,
//            SPACE=(TRK,(5,5,10)),UNIT=unit-name,
//            VOL=SER=volser
```

On completion of the above step a partitioned dataset containing JCL to be used during the remainder of the installation process will have been created.

2) Edit member SMPEDEF from the installation JCL dataset unloaded in the preceding step, as described within the dataset member itself. This job will define and initialize the SMP/E consolidated software inventory (CSI) dataset and create other SMP/E system datasets and the DB2 Manager target and distribution libraries. If any of these datasets are already in existence, delete them prior to running this job or amend the job to remove dataset creation or initialization processing, as required.

   After editing the member, submit it. Verify that all job steps complete with condition code 0, and that all datasets are created and cataloged successfully.

3) Edit JCL member SMPERCV, as described within the member itself. This job performs SMP/E RECEIVE processing for the DB2 Manager V1.1 function SYSMOD.

   After editing the member, submit it. Verify that SMP/E processing has completed with condition code 0.

4) Edit JCL member SMPEAPP, as described within the member itself. This job performs SMP/E APPLY processing for the DB2 Manager V1.1 function SYSMOD.

   After editing the member, submit it. Verify that SMP/E processing has completed with condition code 0.

5) Optionally edit and submit JCL member SMPEACC. This job performs SMP/E ACCEPT processing for the DB2 Manager V1.1 function SYSMOD. Processing of this job may be deferred until later. However, no service can be applied to the product until SMP/E ACCEPT processing has been performed successfully.

The DB2 Manager product distribution libraries will have been successfully created on completion of the above steps. Continue with product installation processing in the following section.

Update DB2 Manager parameter library

The DB2 Manager parameter library unloaded in the previous step will contain sample parameters for the following two DB2 Manager parameter library members:

ENVCNTL parameters which inform DB2 Manager of its operating environment.

TAPECNTL parameters which control the DB2 Manager control region configuration and environment.

After unloading the sample parameter library, review all parameter values, and amend as necessary to reflect the host system's operating environment, and your own requirements for DB2 Manager migration and retrieval processing.

Refer to chapter 3 of this manual for a full description of the DB2 Manager parameter library and the format of individual parameters.

Perform MVS host system modifications

The following MVS system changes will be required in order to implement DB2 Manager:

a. APF-authorize DB2 Manager and Archive Manager load libraries.

The DB2 Manager load library (file 4 from the distribution tape) and the Archive Manager load library (created during the Archive Manager installation process) should both be APF-authorized (APF-authorization is optional for stand-alone Archive Manager operation). Refer to IBM system documentation for a description of how to perform this procedure on your system.

b. Create DB2 Manager control region procedure.

The JCL member LCDDB2 from the distributed JCL library (file 3 on the distribution tape) should be copied to a system procedure library and amended as appropriate. The member and procedure name may be modified as required. Update the procedure with the names of the load and parameter libraries created during the distribution tape unload process. A listing of the distributed LCDDB2 procedure is given in appendix A.

The LCDDB2 procedure must be started in order to initiate DB2 Manager control region processing. Note that the control region started task may be given any name chosen during install processing. The procedure will be referred to as LCDDB2 throughout the remainder of this manual.

c. Assign DB2 Manager user identification.

A unique system user identification (eg. LCDDB2) should be assigned for DB2 Manager use, using whatever system security product is installed on the host system. This user id should be assigned to execution of the DB2 Manager control region,.

The DB2 Manager user id may be used to control access to protected system resources, as determined by the system security administrator.

d. Update the system Program Properties Table (PPT).

The PPT should be updated in order to identify the DB2 Manager control region program OTDBP200 as a non-swappable system task. Refer to IBM system documentation for a description of how to perform this procedure on your system.

The following is an example of how to specify the PPT entry in the appropriate SCHEDxx member of SYS1.PARMLIB:

```
PPT         PGMNAME(OTDBP200)
            NOSWAP
            KEY(7)
            SYST
```

e. Update FLPA or MLPA definitions.

The DB2 Manager SQL intercept module OTDBP300 from the distribution LPA library must be installed in the FLPA or MLPA. There are no functional considerations for DB2 Manager between installation of this module in the FLPA or MLPA, and consequently the choice may be left to the system programmer responsible for the administration of the system on which the installation is being performed.

Using the FLPA will eliminate paging for execution of module OTDBP300, but will reduce the total amount of real storage available for system use by about 2k bytes. If the amount of real storage available is limited, allow the above module to be paged as normal by installing it in the MLPA.

To install in the FLPA, perform the following tasks:

- Add module OTDBP300 to member IEAFIXxx in SYS1.PARMLIB. OTDBP300 is on the distributed DB2 Manager LPA library – this library should be specified in the INCLUDE parameter as the location of the required module, or the module copied to another library which already has an entry in the SYS1.PARMLIB member.
- Add the NOPROT option to the FIX parameter in the IEASYSxx member in SYS1.PARMLIB, as follows:

```
FIX = (xx,NOPROT)
```

To install in the MLPA, perform the following tasks:

- Add module OTDBP300 to member IEALPAxx in SYS1.PARMLIB. OTDBP300 is on the distributed DB2 Manager LPA library – this library should be specified in the INCLUDE parameter as the location of the required module, or the module copied to another library which already has an entry in the SYS1.PARMLIB member.
- Add the NOPROT option to the MLPA parameter in the IEASYSxx member in SYS1.PARMLIB, as follows:

```
MLPA = (xx,NOPROT)
```

f. Allocate DB2 Manager SMF record identifier.

The DB2 Manager control region will optionally create SMF records during object retrieval processing. This option is controlled via the SMFRECID parameter in the ENVCNTL parameter library member.

If it is required that DB2 Manager should create SMF records, a unique SMF user record identifier (in the range 128-255) should be assigned for DB2 Manager use. The allocated value should then be identified to DB2 Manager via the ENVCNTL SMFRECID parameter.

g. Perform tape management system changes.

Primary tape datasets created by DB2 Manager will all have a high-level qualifier of

```
'hlq.G*'
```

where ‘hlq’ is the high-level qualifier defined for the Archive Manager database containing the tape volume (see page 27).

Backup or duplex tape datasets (as created during object migration) will all have high-level qualifiers of

```
'hlq.xn.G*'.
```

where ‘hlq’ is the high-level qualifier defined for the Archive Manager database for the storage group containing the tape volume (see page 27), ‘x’ is the copy identifier, and ‘n’ is the Archive Manager storage level (‘0’ or ‘1’). The copy identifier ‘x’ may take the following values:

B - indicates the primary backup copy created by the Archive Manager database backup utility OTASP110.

C - indicates the duplex copy created during the object storage process (by table migration utility OTDBP100). This copy will only be created if Archive Manager duplexing has been enabled for the database being updated.

D - indicates the duplex backup copy created by the Archive Manager database backup utility OTASP110. This copy will only be created if Archive Manager backup duplexing has been enabled for the database being updated.

Any tape management software present on the system should be amended as required to establish appropriate scratch pool and retention period controls. If required, the retention period or expiry date of tape volumes created in an Archive Manager database may be specified via the Archive Manager database administration procedure, in TSO/ISPF.

Where possible, a retention period of ‘CATALOG’ should be established for each DB2 Manager scratch pool. When used with the Archive Manager ‘AUTOUNCAT’ facility, this will facilitate automatic return of tape volumes to the scratch pool after release by the Archive Manager database maintenance utility. Refer to the Archive Manager User Manual for further information about tape scratch pools and the tape database maintenance utility.

Primary or secondary DB2 Manager tape datasets will be released under the following circumstances:

- The Archive Manager tape backup utility will automatically uncatalog backup datasets which are being recreated during the migration process (this will occur when objects are added to a partially-used volume during the migration process; the backup of the partially-used volume will be released).
- The Archive Manager tape database management utility will release volumes from within a storage group database which contain less than a user-supplied threshold of active objects. Released volumes will be automatically uncataloged by the utility if the AUTOUNCAT option has been specified for that storage group database via the Archive Manager database maintenance facility.

h. Perform STK Host Software Component (HSC) changes.

The relevant modifications should be made to the HSC software which controls operation of the StorageTek library configuration and environment, in order to ensure that scratch tapes used during the DB2 Manager object migration process are located within the library configuration. All Archive Manager tape datasets will have a high level qualifier as specified during the Archive Manager database definition procedure (see page 27).

It is essential for object retrieval processing that all primary volumes within an Archive Manager database can be mounted automatically in response to an object recall request. For this reason, all such cartridges should be located within a StorageTek library configuration.

Appropriate controls should also be established to handle creation of backup and duplex tape datasets as required (eg) use scratch tapes from non-library drives with automatic cartridge loaders (for subsequent storage on library shelving), or use scratch tapes from within a remote library configuration (for automatic offsite storage).

Refer to the relevant StorageTek manual, or contact your StorageTek Software Support Representative (SSR) for details on how to implement the above controls.

i. Update automated reply software controls.

Occasionally, it may happen that there are no tape drives available to satisfy a request from DB2 Manager for object retrieval from a non-mounted tape volume.

When this occurs, normal allocation recovery processing will be performed by the host system control program, resulting in the display of message IEF247I followed by the following message:

```
IEF238D jjj - REPLY DEVICE NAME, 'WAIT' OR CANCEL'
```

While this reply is outstanding, DB2 Manager will be unable to service the request which caused allocation recovery to be entered.

To avoid any unnecessary and unpredictable delays due to this situation, any automated operator reply facility which is present on the host system should be configured to reply 'CANCEL' to the above message for any DB2 Manager retrieval request (ie) a request for retrieval from an Archive Manager tape dataset.

DB2 Manager will handle a reply of 'CANCEL' in the following manner:

1. For row retrieval requests, a ‘resource unavailable’ response will be raised by DB2 Manager. An SQL –652 error is then returned to the caller and a reason code placed field SQLERRD(6) in the SQLCA (refer to chapter 7 of this manual for a description of these codes), or optionally the request will be held internally in DB2 Manager until a tape drive becomes available or until a customer-specified time interval has elapsed (at which time the request will be rejected as above). For rejected requests, the calling application may be amended to handle this type of error in a special manner (eg) to display a retry message to the end user.
2. For row migration requests, DB2 Manager will retry the allocation at one-minute intervals, up to a retry limit of 30. If, after this limit has been reached, allocation has still not been successful, an error is returned to the DB2 Manager migration utility, and the program will terminate.

Perform DB2 system modifications

DB2 Manager requires its own DB2 database for execution. This database must be defined in all DB2 systems which are to be used by DB2 Manager and the appropriate authorities granted before the product can be used. Each DB2 system must also be updated with details of the plan(s) required by DB2 Manager. The following activities will need to be performed:

a. Create LCDDB2 system database and grant access authorities.

DB2 Manager uses its own database for retention of internal information. This is a DB2 database containing the following tables:

- LCDDB2.REGISTRY. This table is used for registration of tables which have been enabled for row migration, and identifies the Archive Manager database used for storing migrated rows from each of the tables. It is automatically maintained during DB2 Manager migration processing.

- LCDDB2.PLAN_TABLE. This table is used internally by DB2 Manager for DB2 EXPLAIN processing when validating customer-specified row selection criteria.

During normal operation, DB2 Manager will need to access and update resources in its system database. The following table identifies these database resources and the level of access required to each.

Resource type	*Resource name*	*Access required*
DATABASE	LCDDB2	CREATETAB
TABLESPACE	LCDDB2.LCDDB2TS	USE OF

TABLE	LCDDB2.PLAN_TABLE	SELECT,DELETE,
		INSERT,UPDATE

TABLE	LCDDB2.REGISTRY	SELECT,DELETE,I NSERT,UPDATE

Member OTDBDEF in the distributed DB2 Manager JCL library supplies a job to allocate and define this database, and to grant the above authorities. Edit this member, following the instructions at the head of the member, and submit the job. A copy of the supplied OTDBDEF job can be found in appendix A of this manual.

b. Create DB2 Manager application plan.

DB2 Manager requires one or more DB2 plans for execution of its batch utilities. Plan names are specified via the PLAN execution parameter for each utility.

It should be noted in this context that DBRMs shipped with the base product or with service packages are created with the VERSION(AUTO) pre-compiler option. This will allow multiple copies of the same package to be bound to a single DB2 plan, to allow different versions of DB2 Manager programs to execute using that plan. Alternatively, separate plans can be created and execution controlled via use of the PLAN execution parameter.

Edit member OTDBPKG in the distributed DB2 Manager JCL library (file 3 on the distribution tape). This contains JCL to create DB2 packages from DBRMs supplied with the product. Instructions for editing the JCL are provided at the top of the file.

After executing member OTDBPKG, edit and submit member OTDBPLAN in the distributed DB2 Manager JCL library (file 3 on the distribution tape). This contains JCL to create the application plan to be used by DB2 Manager and to grant DB2 authorities for access to the plan. Again, instructions for editing the JCL are provided at the top of the file.

A copy of the supplied OTDBPKG and OTDBPLAN jobs can be found in appendix A of this manual.

Define Archive Manager databases

DB2 Manager uses the Archive Manager component of the Lifecycle Director of DB2 product for storage and retrieval of objects. Migrated rows are held in one or more Archive Manager databases, one Archive Manager database being used per DB2 table which has been enabled for migration processing (ie) there is a one-for-one correspondence between DB2 tables and Archive Manager databases in the DB2 Manager configuration. This one-for-one correspondence is enforced during row migration processing.

Each Archive Manager database must be defined prior to first migrating rows from a DB2 table to DB2 Manager.

Definition of an Archive Manager database is performed using the standard Archive Manager database administration dialogs under TSO/ISPF. Refer to the Archive Manager User Manual for a full description of the Archive Manager database definition procedure.

This section summarizes the procedure for the definition of an DB2 Manager migrated row database.

1. Logon to the Archive Manager database administration dialog in TSO/ISPF. Consult the person(s) responsible for Archive Manager database administration in your installation if you are unsure of how to invoke this facility.
2. From the main Archive Manager database administration menu, enter the name to be assigned to the database's primary index dataset in the 'primary index name' field. Any valid unique name may be chosen for this dataset.

   Select option 3 from this menu, and press ENTER. Panel OTSN4300 (the base database definition panel) will be displayed.
3. Fields on panel OTSN4300 should be entered as follows (see Figure 1):

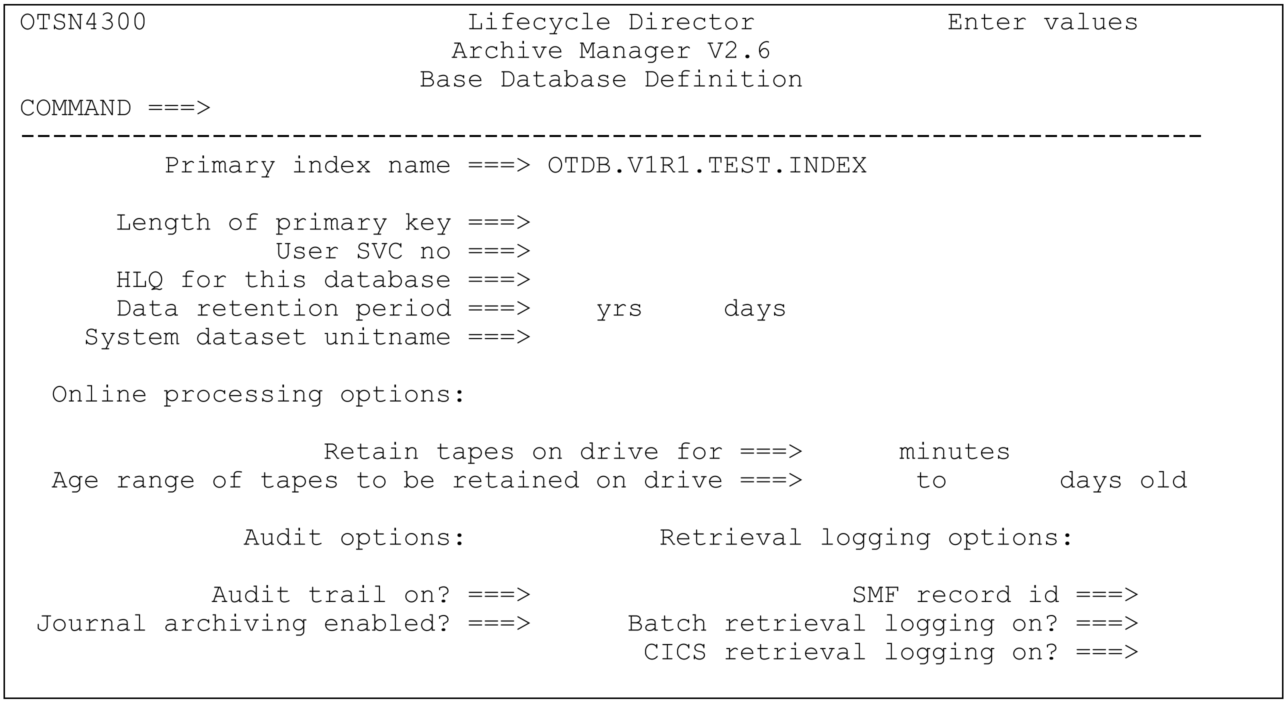
   Length of primary key — *Enter the value '4'.*

   Archive Manager user SVC no — Enter the user SVC number allocated for Archive Manager during the product installation procedure. This must be a numeric value in the range 200-255. Consult the person(s) responsible for installing Archive Manager on your system if you are unsure of this value.

   HLQ for this database — Enter the 1-30 character high level qualifier to be assigned by Archive Manager for all system datasets created within the database.

Data retention period

This entry should be allowed to default to a value of 0 years and 0 days (ie) no automatic expiry. Archive Manager objects should not be allowed to expire. They will be explicitly deleted by the DB2 Manager database housekeeping utility when they do not contain any more active migrated rows.



```
OTSN4300                    Lifecycle Director            Enter values
                           Archive Manager V2.6
                         Base Database Definition
COMMAND ===>
--------------------------------------------------------------------------
         Primary index name ===> OTDB.V1R1.TEST.INDEX

      Length of primary key ===>
                User SVC no ===>
      HLQ for this database ===>
      Data retention period ===>    yrs     days
    System dataset unitname ===>

  Online processing options:

                   Retain tapes on drive for ===>      minutes
  Age range of tapes to be retained on drive ===>       to       days old

              Audit options:            Retrieval logging options:

            Audit trail on? ===>                    SMF record id ===>
 Journal archiving enabled? ===>      Batch retrieval logging on? ===>
                                       CICS retrieval logging on? ===>
```

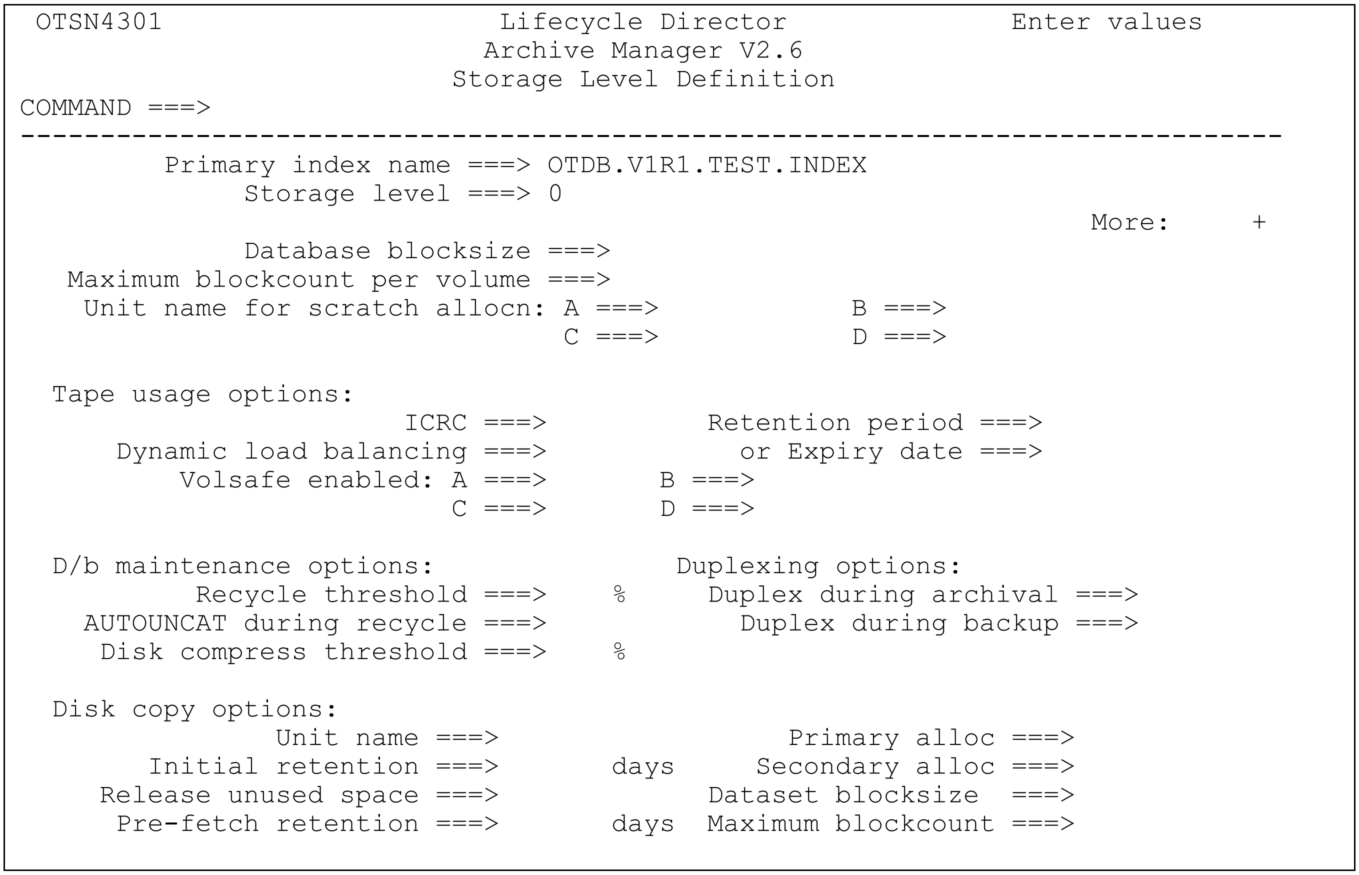
Figure 1 - base database definition panel OTSN4300

Online processing options

These fields should be left blank. Entries in these fields are only used by the CICS retrieval interface in Archive Manager. This interface is not used by DB2 Manager.

The ‘audit options’ fields are optional, and may be entered as required. Refer to the Archive Manager User Manual for a description of these fields and their usage.

After all required entries have been made, press 'ENTER'. When all entries have been accepted, the storage level definition panel OTSN4301 will be displayed to allow storage level 0 to be defined for the Archive Manager database.



```
 OTSN4301                     Lifecycle Director             Enter values
                             Archive Manager V2.6
                           Storage Level Definition
COMMAND ===>
-------------------------------------------------------------------------------
         Primary index name ===> OTDB.V1R1.TEST.INDEX
              Storage level ===> 0
                                                                   More:     +
              Database blocksize ===>
   Maximum blockcount per volume ===>
    Unit name for scratch allocn: A ===>            B ===>
                                  C ===>            D ===>

  Tape usage options:
                        ICRC ===>          Retention period ===>
      Dynamic load balancing ===>            or Expiry date ===>
          Volsafe enabled: A ===>       B ===>
                           C ===>       D ===>

  D/b maintenance options:               Duplexing options:
           Recycle threshold ===>    %     Duplex during archival ===>
    AUTOUNCAT during recycle ===>            Duplex during backup ===>
     Disk compress threshold ===>    %

  Disk copy options:
                Unit name ===>                  Primary alloc ===>
        Initial retention ===>       days     Secondary alloc ===>
     Release unused space ===>             Dataset blocksize  ===>
      Pre-fetch retention ===>       days  Maximum blockcount ===>
```

Figure 2 - storage level definition panel OTSN4301

4. Fields on panel OTSN4301 should be entered as follows (see Figure 2):

Database blocksize Enter the value ‘32760’.

Unit name for scratch allocation Enter the unit name(s) to be used by Archive Manager when dynamically allocating primary and backup copies of new tape datasets in storage level 0 in this Archive Manager database. These should be a 1 to 8-character alphanumeric identifier which identify a valid esoteric unit name or device address on the host system control program.

All other fields on this panel are optional. Refer to the Archive Manager User Manual for a description of these fields and their usage.

After all required entries have been made, press 'ENTER'. When all entries have been accepted, the primary index Access Method Services panel OTSN4302 will be displayed to enable customization of the primary index definition parameters for the Archive Manager database being defined.

5. After all database definition and initialization parameters have been accepted, one or more Access Method Services (AMS) control panels will be displayed to allow customization of AMS parameters used by Archive Manager when defining system datasets for the database being defined.

AMS control panels will be displayed in the following sequence:

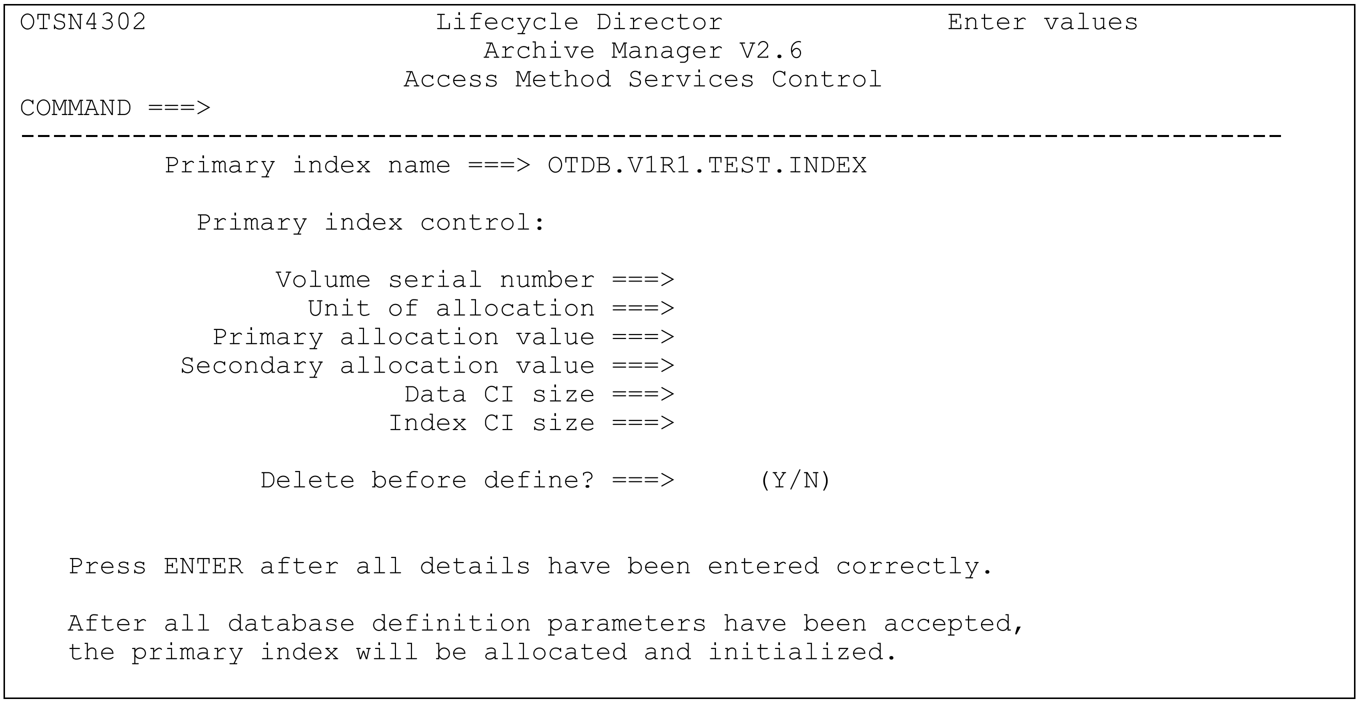
1. **OTSN4302** - the primary index AMS control panel. This panel is always displayed. Values entered on this panel will be used when creating the primary index dataset (VSAM KSDS) for the database being defined.
2. **OTSN4303 -** the journal dataset Archive Manager control panel. This panel will only be displayed if the audit control option was enabled on panel OTSN4300. Values entered on this panel will be used during creation of the journal dataset (VSAM ESDS) for the database being defined.
3. **OTSN4304 -** the journal archive database AMS control panel. This panel will only be displayed if both the audit control facility and journal archiving were enabled on panel OTSN4300. Values entered on this panel will be used during creation of the journal archive database primary index dataset (VSAM KSDS).

Panel OTSN4302 is illustrated in Figure 3. Each of the above panels will contain some or all of the following fields:

Volume serial number Optional entry. Enter the serial number (1-6 alphanumeric, national or special characters) of the disk volume on which the dataset is to be created.

This parameter may be omitted if the volume is to be automatically selected by the operating system.

Unit of allocation *For primary index definition (panel OTSN4302) enter a value of 'CYL'. For journal or journal archive definitions (panels OTSN4303 and panel OTSN4304), refer to the Archive Manager User Manual for a discussion of sizing of these datasets.*



```
OTSN4302                  Lifecycle Director             Enter values
                             Archive Manager V2.6
                        Access Method Services Control
COMMAND ===>
-------------------------------------------------------------------------------
         Primary index name ===> OTDB.V1R1.TEST.INDEX

           Primary index control:

                Volume serial number ===>
                  Unit of allocation ===>
            Primary allocation value ===>
          Secondary allocation value ===>
                        Data CI size ===>
                       Index CI size ===>

               Delete before define? ===>     (Y/N)


   Press ENTER after all details have been entered correctly.

   After all database definition parameters have been accepted,
   the primary index will be allocated and initialized.
```

Figure 3 - Access Method Services control panel OTSN4302

Primary allocation value For primary index definition (panel OTSN4302), enter the number of cylinders to be used for allocation of the primary data extent of the index dataset.   Each object containing migrated rows will occupy 40 bytes of index storage. The maximum number of migrated rows is set using the OBJSIZE parameter in the ENVCNTL parameter library member. Ensure that the primary index allocation value is sufficient to contain the highest number of objects that are expected to be held in this Archive Manager database.

For journal or journal archive definitions (panels OTSN4303 and panel OTSN4304), refer to the Archive Manager User Manual for a discussion of sizing of these datasets.

Secondary allocation value Enter the required secondary allocation value for the index or journal datasets, as appropriate.

Data CI size Enter the control interval (CI) size to be used for allocation of the data component of the dataset. This should be a valid VSAM control interval size. If omitted, a control interval size of 16k is used for the data component.

Index CI size Optional entry. Enter the control interval size to be used for allocation of the index component of the dataset (KSDS only). This should be a valid VSAM control interval size. If omitted, a control interval size of 2k is used for the index component.

Delete before define? Optional entry. Enter the value 'Y' if Archive Manager should attempt to delete the dataset before it is defined. Enter 'N' if no deletion request should be issued.

Press 'ENTER' after all required entries have been made on each panel. After all AMS control panel entries have been accepted, the database will be defined and initialized.

The above Archive Manager database definition procedure should be executed once for each DB2 table which is to be processed by DB2 Manager.

■ Verify DB2 Manager installation

After all DB2 Manager installation activities, as specified in the preceding sections, have been successfully performed, product implementation may be verified by starting the DB2 Manager control region.

Before doing this, ensure that all necessary parameters in the ENVCNTL and TAPECNTL parameter library members have been coded correctly, as described in chapter 3.

The DB2 Manager control region may be started by entering the command:

```
START procedure_name
```

on the system console.

If ‘COMMAND=REPLY’ has been specified (or allowed to default) in the ENVCNTL parameter library member, the operator reply message:

```
OTD20000 Enter request:
```

will be displayed on the console after successful initialization of the control region. Operator commands may be entered by replying to the outstanding operator reply message.

If ‘COMMAND=MODIFY’ has been specified in the ENVCNTL parameter library member, operator commands may be entered using the MVS MODIFY command structure.

Refer to chapter 4 of this manual for details of the operator commands available to control and shutdown the DB2 Manager control region.

If any error occurs during control region initialization processing, refer to the description of the displayed message in chapter 7 for details of the initialization error. Perform the appropriate corrective action before restarting the DB2 Manager control region.

DB2 Manager Parameter Specification

3

DB2 Manager uses its own parameter library to allow users to control the operation of the product. This parameter library is a partitioned dataset containing the following members:

ENVCNTL - used to specify general parameters relating to the DB2 Manager operating environment.

TAPECNTL - used to specify controls relating to object retrieval processing by the DB2 Manager control region.

A sample parameter library containing all of the above members is distributed with the product on file 4 of the distribution tape. This file will provide a sample specification of each of the parameters used by DB2 Manager. The sample parameter library may be used for DB2 Manager operation at your installation after updating the library members as required.

Use ISPF EDIT, or any equivalent editing facility present on your system, to create and/or maintain these parameter library members.

DB2 Manager parameters will be validated during control region initialization processing. Any invalid parameter used will be reported via a display in the system and job logs, and initialization of the control region will terminate. In this case, the parameter in error should be corrected on the parameter library, and the DB2 Manager control region restarted.

General parameter format

Each parameter library member will consist of a variable number of 80-byte records, each record containing one or more separate parameter entries. In general, each parameter entry must be completely contained within one record (ie) continuation of any single parameter entry from one record to another is not permitted. A single record may contain multiple parameter entries.

A parameter entry has the general format:

```
keyword = value
```

Each keyword and its associated value are separated by an '=' sign, and optionally one or more spaces. Multiple keyword parameter entries may be specified in each parameter record, each 'keyword=value' string being separated by one or more spaces, and optionally a single comma (',').

The first keyword parameter in a record may begin in any character position from 1 onwards. Parameters may extend up to character position 80 of the record. Individual keyword parameters may not be continued over more than one record.

An asterisk ('*') in character position 1 will indicate a comment record, which will be ignored for parameter validation processing.

Any departure from this general format will be indicated with individual parameter descriptions.

■ ENVCNTL parameters

These parameters are used to pass information to DB2 Manager about its operating environment, and to control certain aspects of DB2 Manager operation which do not fall into the categories controlled by other parameter library members.

Parameters in this member have the general parameter format described on page 34.

ENVCNTL: SMFRECID

```
SMFRECID = 0|nnn
```

Requirement: Optional

Default: 0

Validation: 'nnn' = 3 numeric digits in the range 128-255.

Use: The SMFRECID parameter is used to control the creation of SMF records by DB2 Manager for the recording of migrated row storage and retrieval activities.

A value of 0 indicates that no SMF recording is to be performed by DB2 Manager.

A non-zero value (in the range 128-255) is used to specify the value of the identifier of the SMF records written by DB2 Manager. The default value is 0 (ie) no SMF recording.

Notes: Refer to page 63 for a discussion of DB2 Manager SMF usage, and for a description of the DB2 Manager SMF record layout.

ENVCNTL: OBJSIZE

```
OBJSIZE = nnnnn
```

Requirement: Optional

Default: 1000

Validation: 'nnnnn' = a 1- to 5-digit numeric value in the range 1-65535.

Use: The OBJSIZE parameter is used to specify the maximum number of migrated rows to be stored per Archive Manager object.

Notes: Increasing the maximum number of rows which may be stored per Archive Manager object will reduce the size of the primary index file required for indexing objects in the Archive Manager database, but will add a little overhead to the time taken to retrieved a migrated row from the database.

During row retrieval processing, Archive Manager will first of all locate the start of the object containing the migrated row, using a high-speed block locate process, then process the object sequentially until the logical record containing the migrated row is encountered. A larger number of migrated rows (i.e. logical records) per object will increase the average time taken to perform this sequential record locate process.

Note that the value specified (or defaulted) here for the global OBJSIZE parameter will be used in conjunction with the OTDBP100 COMMIT parameter specified during table migration processing, in order to determine the number of rows to be stored per object in the Archive Manager database. The actual number of rows per objects will be the lower of the two values (COMMIT or OBJSIZE).

ENVCNTL: READTIMEOUT

```
READTIMEOUT = nnnn
```

Requirement: Optional

Default: 10

Validation: ‘nnnn’ = a 1- to 4-digit numeric value in the range 1-1440.

Use: The READTIMEOUT parameter specifies the maximum number of minutes that the SQL intercept routine is to wait for a response from the control region, during row retrieval processing. If no request is received within this time, the request will be rejected with a ‘timeout’ condition.

An SQL error of –652 will be returned to the requesting application.   Field SQLERRD(6) in the SQLCA will contain the hexadecimal value x’0A’.

ENVCNTL: WRITETIMEOUT

```
WRITETIMEOUT = nnnn
```

Requirement: Optional

Default: 10

Validation: ‘nnnn’ = a 1- to 4-digit numeric value in the range 1-1440.

Use: The WRITETIMEOUT parameter specifies the maximum number of minutes a writer task is to wait for a request, when connected to an Archive Manager database.   If no request is received within this time, the writer task will issue an Archive Manager ‘disconnect without commit’ request to disconnect from the Archive Manager database.

Any subsequent request from the table migration utility to store a migrated row in that database will be rejected, and the utility will terminate abnormally.

■ TAPECNTL parameters

These parameters are used to control operation of the DB2 Manager control region.

Parameters in this member have the general parameter format described on page 34.

TAPECNTL: MAXTRDR

```
MAXTRDR = nnn
```

Requirement: Optional

Default: 4

Validation: nnn = 1-3 digit numeric value in range 0 - 255

Use: The MAXTRDR parameter specifies the number of active tape-reader subtasks which are to be created in the DB2 Manager control region.

Notes: This parameter indirectly controls the maximum number of tape drives which may be concurrently allocated by the control region.

If TAPEWAIT=0 is specified, requests which cause this maximum to be exceeded will receive a “resource busy” condition from DB2 Manager - an SQL error code of –652 will be returned to the calling application and field SQLERRD(6) in the SQLCA will contain the hexadecimal value x’0A’.

Non-zero values of TAPEWAIT will cause requests which encounter this condition to be held internally for processing until a tape drive becomes available, or until the time limit specified in the TAPEWAIT parameter is exceeded, at which point a “resource busy” condition will be returned to the calling application, as above.

This setting may be varied during DB2 Manager operation via the 'SET MAXTRDR’ operator command.

If this parameter is omitted, a default value of MAXTRDR=4 will be used.

TAPECNTL: MAXDRDR

```
MAXDRDR = nnn
```

Requirement: Optional

Default: 4

Validation: nnn = 1-3 digit numeric value in range 1 - 255

Use: The MAXDRDR parameter specifies the number of active disk-reader subtasks which are to be created in the DB2 Manager control region. This is set during control region initialization, and may be varied thereafter using the ‘SET MAXDRDR’ operator command.

Notes: This parameter indirectly controls the maximum number of Archive Manager ‘K’ copy files which are to be simultaneously allocated by the DB2 Manager control region for retrieval of disk-resident objects from Archive Manager.

Requests for retrieval of an object from an Archive Manager ‘K’ copy file are added to the disk-reader subtask with the shortest queue length. Increasing the number of disk-reader subtasks will reduce the average queue length per subtask, and may improve retrieval response times.

If this parameter is omitted, a default value of MAXDRDR=4 will be used.

TAPECNTL: MAXTWTR

```
MAXTWTR = nnn
```

Requirement: Optional

Default: 4

Validation: nnn = 1-3 digit numeric value in range 1 - 255

Use: The MAXTWTR parameter specifies the number of active tape-writer subtasks which are to created in the DB2 Manager control region.

Notes: Each execution of the DB2 Manager table migration utility will acquire a single tape-writer subtask. The value of the MAXTWTR controls the maximum number of row migration jobs which may execute concurrently.

This setting may be varied during DB2 Manager operation via the 'SET MAXTWTR’ operator command.

If this parameter is omitted, a default value of MAXTWTR=4 will be used.

TAPECNTL: MAXSCHED

```
MAXSCHED = nnn
```

Requirement: Optional

Default: 4

Validation: nnn = 1-3 digit numeric value in range 1 - 255

Use: The MAXSCHED parameter specifies the number of active scheduler subtasks which are to created in the DB2 Manager control region.

Notes: A scheduler subtask is responsible for receiving requests from the SQL intercept module, validating them and passing valid requests to the appropriate component within the DB2 Manager control region for further processing.

Increasing the number of scheduler tasks will reduce the average length of time that it takes to assign a row storage or retrieval request to the appropriate subtask. However, each scheduler subtask introduces some processing overhead, so care should be taken to avoid over-allocation of this setting.

This setting may be varied during DB2 Manager operation via the 'SET MAXSCHED’ operator command.

If this parameter is omitted, a default value of MAXSCHED=4 will be used.

TAPECNTL: MAXQLEN

```
MAXQLEN = nnn
```

Requirement: Optional

Default: 4

Validation: nnn = 1-3 digit numeric value in range 1 - 256

Use: The MAXQLEN parameter specifies the maximum number of retrieval requests which may be simultaneously queued for any single tape-reader subtask in the control region.

Notes: DB2 Manager will queue requests for retrieval from an Archive Manager tape volume which is already being processed, on a 'first-in first-out' basis, up to the maximum level specified in this parameter.

If TAPEWAIT=0 is specified, requests which cause this maximum to be exceeded will receive a “resource busy” condition from DB2 Manager - an SQL error code of –652 will be returned to the calling application and field SQLERRD(6) in the SQLCA will contain the hexadecimal value x’0A’.

Non-zero values of TAPEWAIT will cause requests which encounter this condition to be held internally for processing until a tape drive becomes available, or until the time limit specified in the TAPEWAIT parameter is exceeded, at which point a “resource busy” condition will be returned to the calling application, as above.

This setting may be varied during DB2 Manager operation via the 'SET MAXQLEN' operator command.

If this parameter is omitted, a default value of MAXQLEN=4 will be used.

TAPECNTL: RETAINTAPE

```
RETAINTAPE = nnnn
```

Requirement: Optional

Default: 0

Validation: nnnn = 1-4 digit numeric value in range 0 - 1440

Use: The RETAINTAPE parameter specifies the number of minutes for which a tape is to remain mounted after all outstanding retrieval requests from that tape have been processed by the DB2 Manager control region.

A tape which has not been referenced for the specified interval will be automatically dismounted. A dismount of a tape may be forced by DB2 Manager prior to expiration of this time interval if no free tape drive is available for allocation of a new tape volume, and the tape is the least-recently referenced of all currently mounted tape volumes.

A value of 0 indicates that tapes will be dismounted immediately after the last outstanding request for retrieval from each tape has been processed.

A value of 1440 indicates that tapes are held permanently on a drive until a dismount is forced when there are no free tape drives to satisfy a new request (as described above), or until termination of the DB2 Manager control region.

Notes: This setting may be varied during DB2 Manager operation via the 'SET RETAINTAPE' operator command.

If this parameter is omitted, a default value of RETAINTAPE=0 will be used.

TAPECNTL: TAPEWAIT

```
TAPEWAIT = nnnn
```

Requirement: Optional

Default: 0

Validation: nnnn = 1-4 digit numeric value in range 0 - 1440

Use: TAPEWAIT is an optional parameter which is used to control whether retrieval requests for tape-resident Archive Manager objects which cannot be immediately satisfied are rejected (with a ‘resource unavailable’ reason code) or queued internally until a tape retrieval resource becomes available, or until a maximum wait limit has been exceeded.   ‘nnnn’ must be a 1-4 digit numeric character string in the range 0-1440, which specifies the maximum number of minutes for which requests are to be queued internally.

If set to 0 (the default), no internal queuing of requests will take place in the above circumstances and the request will be rejected with a ‘resource unavailable’ reason code.

If the TAPEWAIT parameter value is in the range 1-1439, requests will be queued internally until processed or until the maximum queuing time in minutes (as specified in this parameter) has been exceeded. Requests will then be rejected with a ‘resource unavailable’ reason code (as above).

If the TAPEWAIT parameter value is 1440, then requests will be held indefinitely in the internal queue until the necessary resources to process the request become available.

Notes: This setting may be varied during DB2 Manager operation via the 'SET TAPEWAIT’ operator command.

If this parameter is omitted, a default value of TAPEWAIT=0 will be used.

TAPECNTL: COMMAND

```
COMMAND = REPLY|MODIFY
```

Requirement: Optional

Default: COMMAND=REPLY

Validation: The parameter values REPLY or MODIFY must be specified.

Use: This is an optional parameter which is used to control the method by which operator commands are passed to the DB2 Manager control region. The default value of ‘REPLY’ will cause the control region to output the operator reply message OTD20000 when it is ready to receive an operator command.

If ‘MODIFY’ is specified, no operator reply message will be displayed. Operator commands must be entered via an MVS MODIFY command

(eg) `F LCDDB2,SET MAXTRDR=2`

The value of this setting cannot be modified during control region operation

This page is intentionally left blank

DB2 Manager Control Region

4

All access to Archive Manager databases containing migrated rows is performed via the DB2 Manager control region.   This is a started task, which must be initiated prior to performing any DB2 row migration or retrieval activities.

This chapter describes the operation and control of the DB2 Manager control region.

The DB2 Manager control region performs the following functions:

- Enables DB2 Manager for Archive Manager database processing.
- Enables interception of all SQL insert, update and retrieve commands for DB2 tables which have been enabled for DB2 Manager migration processing.
- Establishes the environment for object storage and retrieval from Archive Manager.
- Processes SQL retrieve and update requests for rows which have been migrated to Archive Manager.
- Performs object storage processing when migrating rows to an Archive Manager database.
- Performs SMF record creation to record DB2 Manager row access activity.
- Processes requests from the operator interface facility.

No other system activities are required in order to enable DB2 Manager row storage and retrieval processing. In particular, no modifications are required to existing application coding to allow access to rows which have been migrated to an Archive Manager database.

■ Control region initialization

The DB2 Manager control region is initiated as an MVS started task. Any valid procedure name may be assigned to this started task. The procedure used for starting this task was created during the DB2 Manager installation process. JCL to execute this procedure is supplied in the sample JCL library distributed with the product.

The following points should be noted regarding creation and maintenance of the LCDDB2 procedure:

- The distributed DB2 Manager and Archive Manager load libraries must be available to this task, either via the system linklist, or from the procedure's STEPLIB concatenation.
- The DB2 Manager control region must run with APF-authorization. Both the DB2 Manager and Archive Manager load libraries must therefore be APF-authorized. If these libraries are specified via the STEPLIB concatenation, failure to APF-authorize either of them will result in the loss of APF-authorization for the control region itself. This is likely to cause a system 047 abend during control region processing.
- The installed DB2 Manager parameter library should be specified in the procedure's SYSPARM DD card. Parameter settings should be reviewed and set as required prior to control region start-up (see chapter 3).
- The primary index for each Archive Manager database containing rows which have been migrated from DB2 must be identified in a DD entry in the control region procedure.   Each entry should have a DD name of OTASXXXX (where XXXX is the 4-character identifier of the Archive Manager database, as specified in the execution parameters of the table migration utility – see page 82), and specify the dataset name of the primary index of the Archive Manager database (e.g.)

```
//OTASST01 DD DSN=Archive Manager.ST01.INDEX,DISP=SHR
```

To start DB2 Manager control region initialization, enter the system operator command – note that a procedure name of LCDDB2 is used in the following illustrations:

```
START LCDDB2
```

The DB2 Manager control region will display a number of informational messages during initialization processing. After successful initialization, if ‘COMMAND=REPLY’ was specified (or defaulted) in the TAPECNTL parameter library member the following operator reply message will be displayed:

```
OTD20000 Enter request:
```

If ‘COMMAND=MODIFY’ was specified, the following message will be displayed:

```
OTD20021 Initialization completed successfully
```

DB2 Manager has now initialized successfully, and support has been enabled for access to rows which have been migrated from DB2 to Archive Manager.

Any errors encountered during the initialization process will be displayed on the system console, and DB2 Manager initialization terminated. The appropriate message in chapter 7 of the user manual should be consulted to identify the cause of the error. Take the recommended action to rectify the problem then restart the control region.

Note that DB2 does not need to be started for successful initialization of the DB2 Manager control region to take place.

Control region components

The control region must be initialized in order to enable DB2 Manager to access migrated rows, and to allow migration of rows from DB2 to Archive Manager.

Figure 4 illustrates the structure of the control region and the migrated row access mechanism.   This also shows the MLPA/FLPA-resident SQL intercept processor OTDBP300, which must be defined as a DB2 row edit procedure for each table which is to be enabled for migration processing by DB2 Manager. OTDBP300 will pass any request which requires access to a migrated row to the DB2 Manager control region, and wait for processing of the request to complete before returning control to the calling application.

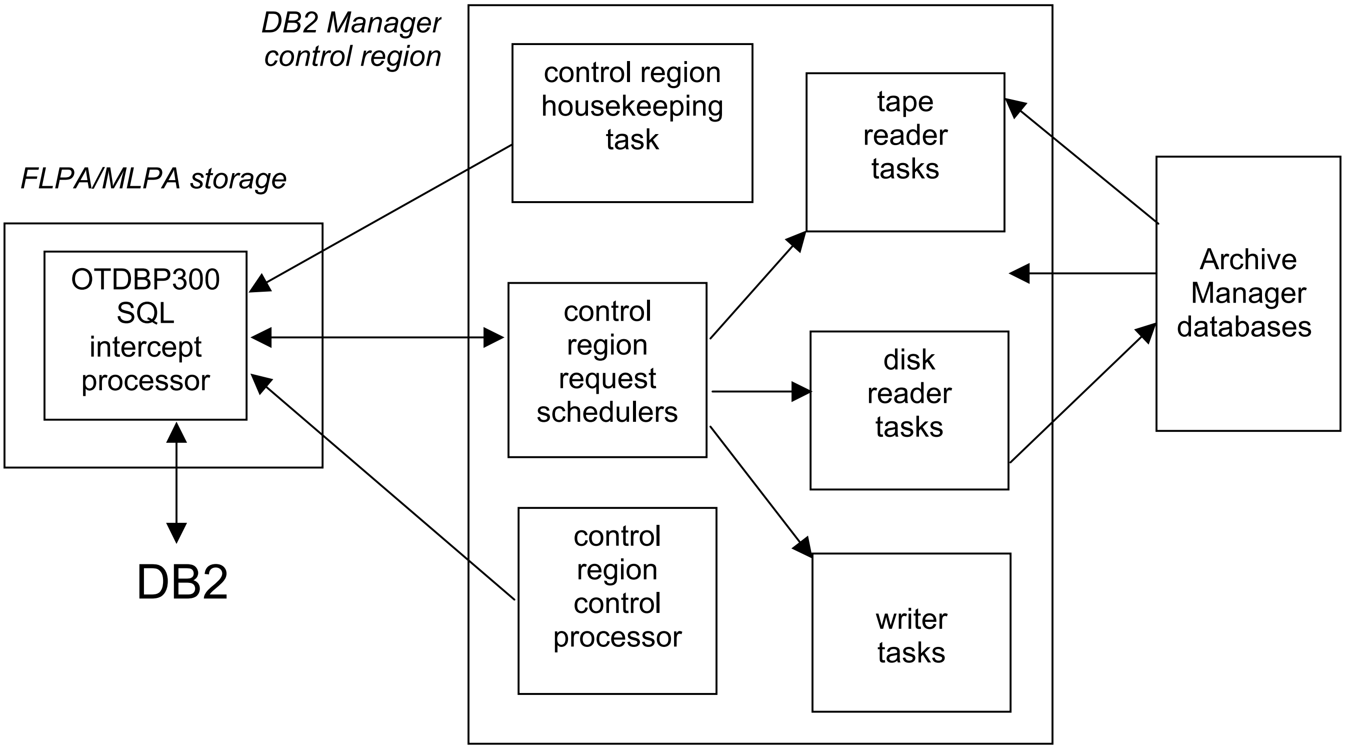

DB2 Manager control region
FLPA/MLPA storage
OTDBP300 SQL intercept processor
DB2
control region housekeeping task
control region request schedulers
control region control processor
tape reader tasks
disk reader tasks
writer tasks
Archive Manager databases

Figure 4: Structure of DB2 Manager control region

The following control region components are illustrated in Figure 4:

- Control processor. This component is responsible for initiation and termination of the control region, and for processing operator requests.
- Writer tasks. There will be one or more writer tasks – the number of writer tasks is controlled via the MAXTWTR initialization parameter. A single writer task will be used exclusively for storage of migrated rows in an Archive Manager database, during execution of the DB2 Manager table migration utility. The existence of multiple writer tasks will enable multiple row migration jobstreams to be executed concurrently on different Archive Manager databases.
- Tape reader tasks. There will be one or more tape reader tasks – the number of these tasks is controlled by the MAXTRDR initialization parameter. A tape reader task will be invoked by the DB2 Manager SQL intercept processor during object retrieval, update and deletion processing, when a migrated row is to be retrieved from a tape-resident Archive Manager object. Each reader task will be able to process a single tape volume at any one time. The number of tape reader tasks controls the number of tape volumes which may be processed concurrently for this type of processing.
- Disk reader tasks. There will be one or more disk reader tasks – the number of these tasks is controlled by the MAXDRDR initialization parameter. A disk reader task will be invoked by the DB2 Manager SQL intercept during object retrieval, update and deletion processing, when a migrated row is to be retrieved from a disk-resident Archive Manager object. The number of disk reader tasks controls the number of disk retrieval requests which may be processed concurrently.
- Scheduler tasks. There will be one or more scheduler tasks – the number of these tasks is controlled by the MAXSCHED initialization parameter. A scheduler task is responsible for validating requests received from the DB2 Manager SQL intercept processor, and assigning the request to an appropriate reader or writer task. Increasing the number of scheduler tasks will reduce the length of time a request waits for validation and assignment processing, but will increase the processing overhead associated with the control region.
- Housekeeping task. A single housekeeping task will be initiated during control region initialization processing. It is responsible for monitoring outstanding requests, and terminating stalled requests.

Processing requests for access to migrated rows

Once DB2 Manager initialization has been completed, batch, TSO, CICS or IMS programs will be able to access migrated rows using standard SQL processing, without modification.

DB2 Manager implements support for access to migrated rows which is made via SQL SELECT, UPDATE and DELETE commands. Note that any rows created in a table via an SQL INSERT command will be stored in DB2. It is not possible to directly store a row in an Archive Manager database – a row must be migrated from a DB2 table to an Archive Manager database, using the DB2 Manager table migration utility.

SQL SELECT processing

Any reference to a migrated row generated via an SQL SELECT command will cause the DB2 Manager SQL intercept processor to pass a request to the DB2 Manager control region. This request will be processed by a scheduler task, which will perform some initial validation.

If the object containing the required row is disk-resident, the scheduler will pass the request to the disk reader task with the shortest request queue.

If the object containing the required row is tape-resident only, the scheduler will pass the request to the most appropriate tape reader task. This is selected using the following hierarchy:

1. A busy task which is processing the tape volume containing the object to be retrieved, and whose request queue is not greater than the maximum specified in the MAXQLEN initialization parameter.
2. A waiting task which has no tape mounted.
3. A waiting task which currently has a tape mounted.

If there is no tape reader task satisfying any of the above conditions, the request will be rejected with a “resource unavailable” condition, unless a non-zero value is specified for the TAPEWAIT initialization parameter. In these circumstances, the request is queued internally and retried until it has been successfully assigned to a tape reader task, or until the time interval specified in the TAPEWAIT parameter has been exceeded, in which case it will be rejected with a “resource unavailable” condition. Refer to page 41 for a description of the TAPEWAIT parameter.

When a request has been passed to a tape or disk reader task, the task will locate the object containing the migrated row and extract the logical record containing the row. This will be passed back to the SQL intercept routine, which will in turn pass the required row back to the calling application.

The RETAINTAPE initialization parameter may be used to retain a tape volume on a drive for a specified period of time when a tape reader task has processed all outstanding requests for retrieval of objects from that tape volume. This will improve response times for subsequent requests to the same tape volume, but may cause an additional processing overhead if the tape has to be dismounted prior to the task allocating a different tape volume (as required in task selection criterion (3) above). Refer to page 40 for a description of the RETAINTAPE parameter.

Note that the above retrieval processing is performed for each migrated row which is referenced via an SQL SELECT command. Where multiple migrated rows are referenced via a single SQL SELECT command and a program cursor used to process the resulting result table, each SQL FETCH command referencing a migrated row will cause a retrieval request to be passed to the control region. This may generate substantial tape handling activity, and result in extended response times in order to access all referenced rows.

On completion of the SQL command, a standard SQL return code will be passed back to the program indicating the success of the request. This can be checked by the program using standard SQL error handling. A return code of 0 indicates that the retrieval request was performed successfully. Any error encountered during the DB2 Manager migrated row retrieval process will generate an SQL error code of –652. Fields SQLERRD(6) and SQLERRD(7) in the SQLCA will contain additional codes indicating the cause of the error.

SQL UPDATE processing

A request from an application to update a migrated row (via an SQL UPDATE request) will cause the row to be retrieved from the Archive Manager database as described for SQL SELECT processing above.

An updated row will then be stored back in the DB2 table (i.e.) the row will be migrated from Archive Manager back to DB2. All reference to the original migrated row will be removed, causing that row to be “orphaned”. The logical record containing this row will be unreferenced. However, the object containing the logical record may contain other active migrated rows, so will continue to exist in the Archive Manager database. The object containing the restored row will be deleted by the DB2 Manager database housekeeping utility when it no longer contains any active migrated rows. Refer to chapter 6 for a description of this utility. Database tape and disk space occupied by the deleted object may be reclaimed, if required, using the Archive Manager object management and database maintenance utilities. Refer to the Archive Manager User Manual for information on this procedure.

Updated rows will become eligible once again for migration from DB2 to Archive Manager, under control of the migration criteria input to the DB2 Manager table migration utility. On migration, the updated row will be written to a different object in the same Archive Manager database as the originally migrated row.

SQL DELETE processing

A request by an application to delete a migrated row via an SQL DELETE command will cause that row to be retrieved via the process described in the preceding section.

The row will then be deleted, causing the archive stub for the row to be removed from the DB2 table. All reference to the migrated row will then have been removed, as for SQL UPDATE processing. Recovery of the space occupied by deleted migrated rows is made in an identical manner to that for updated migrated rows, as described in the preceding section.

SQL error codes

If an attempt to access a migrated row by DB2 Manager is unsuccessful, an SQL return code of –652 is passed back to the calling application program.

A two-byte hexadecimal reason code will be placed in field SQLERRD(6) in the SQL communication area (SQLCA), giving the reason for the failure. In many cases, the control region will also write an error message in the system log.

Standard SQL error handling processing may be used by calling application programs in order to deal with errors encountered during the migrated row access procedure. No changes to existing error handling processing should be required.

Refer to chapter 7 of the user manual for a list of the possible reason codes generated by DB2 Manager during migrated row access processing.

■ DB2 Manager operator interface

DB2 Manager operator commands may be entered by replying to the OTD20000 message on the system console (if COMMAND=REPLY has been specified or defaulted in the TAPECNTL parameter library member), or via an MVS MODIFY command (if COMMAND=MODIFY has been specified in the TAPECNTL member). Only one command may be entered and processed at any one time. When DB2 Manager has completed all processing for a command a message will be displayed on the console indicating the result of the request (and the OTD20000 message re-displayed if COMMAND=REPLY is in effect). Further commands may then be entered by the operator.

Commands are available to perform the following functions:

- Display detailed or summary status of DB2 Manager reader, writer and scheduler tasks.
- Alter the DB2 Manager reader, writer or scheduler task configuration.
- Purge or force purge individual reader, writer or scheduler tasks.

- Terminate the DB2 Manager control region.

All messages issued by DB2 Manager in response to an operator request are documented in chapter 7 of this manual. This section specifies operator request formats, and gives examples of DB2 Manager responses to these requests.

Display summary status

```
D(ISPLAY)
```

This request is used to produce a summary display of the status of the DB2 Manager control region processing configuration.

The DISPLAY command may be abbreviated to the single character 'D'.

DB2 Manager creates a number of tape and disk reader tasks in the control region to perform retrieval of objects from Archive Manager. Each tape reader task is capable of processing one tape volume at any one time. The number of tape reader tasks created is initially equal to the value of the MAXTRDR parameter at DB2 Manager initialization. Increasing this value during DB2 Manager operation will cause a similar increase in the number of tape reader tasks. DB2 Manager will only be able to process as many tape volumes simultaneously as there are initialized tape reader tasks.

Disk reader tasks may process multiple Archive Manager disk (‘K’) copy datasets in one invocation (ie. the task’s request queue may contain requests for retrieval of objects from different disk datasets). DB2 Manager attempts to balance the length of the request queues for the disk reader tasks. Increasing the value of MAXDRDR will reduce the average length of the disk reader task request queues. For most operational environments, the default MAXDRDR value of 4 will be sufficient. Note that the MAXQLEN parameter is not used to control a disk reader task request queue length; there is no limit to the number of requests which may be queued for each disk reader task.

If no disk reader tasks are available (MAXDRDR=0), a retrieval request for a disk-resident object will be passed to an DB2 Manager tape reader task, if one is available. This task will issue a standard Archive Manager object retrieval request, which will still be satisfied wherever possible by Archive Manager from an object’s disk copy. Setting the value of MAXDRDR to 0 in these circumstances is not recommended, as it may cause requests for retrieval of disk-resident objects to be rejected because there are no free tape reader tasks.

Writer tasks are capable of updating a single Archive Manager database at any one time. In addition, an Archive Manager database will not be updated by more than one writer task simultaneously. The number of writer tasks is controlled via the MAXTWTR initialization parameter.

A summary status request will display the current values of the MAXTRDR, MAXDRDR, MAXTWTR, MAXSCHED, MAXQLEN, RETAINTAPE, TAPEWAIT, OBJSIZE, READTIMEOUT and WRITETIMEOUT parameters, the status of the internal scheduler queue lock and the length of the scheduler queue, in message OTD20062.

In addition, message OTD20063 will be displayed for each reader, writer and scheduler task in the system, giving the following details:

Status - this field indicates the current status of the identified task. It can take the following values:

- NULL - the task has never been initialized. It is not available for object recall.
- WAIT - the task is currently waiting for work. For tape reader tasks, no tape volume is currently allocated.
- WTRT - a tape reader task is currently waiting for work, and the last tape volume accessed by the task is still allocated.
- WTSH - a termination request has been made for a waiting task.
- BUSY - the task is currently busy performing object retrieval. For tape reader tasks, the task type will indicate whether the tape volume being processed will be retained on the drive after all outstanding requests have been completed, or whether the tape will be de-allocated and dismounted.
- BYSH - the task is currently busy performing object retrieval, but will terminate when all queued requests have been completed.
- TERM - a terminated task. A task may be terminated either explicitly by operator command, or implicitly via a reduction in the system MAXTRDR, MAXDRDR, MAXTWTR and MAXSCHED settings. It may be restarted by increasing the value of MAXTRDR, MAXDRDR, MAXTWTR or MAXSCHED. A terminated task is not available for object recall or storage.

Qlen - this gives the number of requests queued for processing by this task. All requests will be for retrieval of objects from the tape or disk dataset currently being processed by this task. The queue length will be non-zero for tasks with status = BUSY. For waiting tasks, this value will always be zero.

For busy tasks and waiting tape reader tasks with tape volume still allocated, message OTD20063 may also contain the following information:

DBID - this gives the 4-character identifier of the Archive Manager database currently being processed.

Stg level - identifies the Archive Manager database storage level (in range 0-7) containing the object currently being processed.

Tape - identifies the sequence number within Archive Manager database

Lastuse - for tasks with status = WTRT, this gives the time (in format HH:MM:SS) that the last request for retrieval from this tape volume was completed. This timestamp is used by DB2 Manager to determine which tape volume will be automatically dismounted to satisfy a request for retrieval from another tape volume, when there are no spare drives on the system. The mounted tape volume which was least recently referenced will be dismounted to allow the new request to be processed.

Example:

```
OTD20062 Maxtrdr=004 Maxdrdr=004 Maxtwtr=004 Maxsched=004 Maxqlen=100
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100
OTD20062 Readtimeout=0005 Writetimeout=0005
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=WTRT Qlen=000
DBID=ST01 Stg_level=0 Tape=00002 Lastuse=14:04:52
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=BUSY Qlen=001
DBID=ST02 Stg_level=0 Tape=00005
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=WAIT Qlen=000
OTD20063 Task D002: Status=WAIT Qlen=000
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=BUSY Qlen=001 DBID=PD01
OTD20063 Task W002: Status=WAIT Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
```

Display detail status

```
D(ISPLAY) T(ASK = (T|D|W|S)nnn
```

This request is used to produce a detailed display of the status of the DB2 Manager control region retrieval configuration. The DISPLAY command may be abbreviated to the single character 'D'. The TASK parameter may be abbreviated to the single character 'T'. 'nnn' is the numeric identifier of the task to be displayed (where ‘nnn’ consists of 1-3 numeric characters in the range 1-256). This should be preceded by the character ‘T’ for a tape reader task, ‘D’ for a disk reader task, ‘W’ for a writer task and ‘S’ for a scheduler task. If omitted, a default entry of ‘T’ will be used. This entry must be a valid task number, as displayed via a summary status display request.

A detail status request will first of all display message OTD20063 for the identified task. For busy tasks with a non-zero queue length, a detail status request will display one set of OTD20066 and OTD20067 messages for each request in the queue. These messages will display the following fields:

Request - the entry number of the request in the queue. The request currently being processed will have an entry number of 1.

Type - identifies the type of request queued. This can take the following values:

Open A request to connect to an Archive Manager database for update processing

Retrieve A row retrieval request

Store A row storage request

Commit A request to commit updates to an Archive Manager database

Close A request to disconnect from an Archive Manager database after completion of update processing

DBID - the 4-character identifier of the Archive Manager database being processed by the task

Jobname - the name of the job initiating the request

Stg level - the Archive Manager database storage level being processed by the task. Will be present for storage or retrieval requests only.

Tape - the sequence number of the tape dataset within the Archive Manager database storage level being processed by the task. Will be present for storage and retrieval requests only.

Blockid - the 4-byte identifier of the block containing the object being processed by the task. Will be present for storage and retrieval requests only.

Example:

```
F LCDDB2,D T=T1
......
OTD20063 Task T001: Status=BUSY Qlen=001
DBID=ST01 Stg_level=0 Tape=00002
OTD20066 Request 001: Type=Retrieve DBID=ST01 Jobname=WATERSGJ
OTD20067 Request 001: Stg_level=0 Tape=00002 Blockid=01400005
OTD20058 Display processing completed
```

Force purge task

```
F(ORCE) T(ASK = (T|D|W|S)nnn
```

The FORCE command is used to immediately terminate a tape or disk reader task, writer task or scheduler task in the DB2 Manager control region. Force purging a task does not allow any outstanding requests in the task queue to be completed.

The command may be abbreviated to a single 'F'. The TASK parameter may be abbreviated to a single 'T'. The value 'nnn' must be numeric, and should be preceded by the character ‘T’ for tape reader tasks, ‘D’ for disk reader tasks, ‘W’ for writer tasks and ‘S’ for scheduler tasks. If this character is omitted, a default value of ‘T’ will be used. The task identifier entered here must correspond to a valid task number, as displayed by the status summary command.

Both waiting and busy tasks are immediately terminated. For busy tasks, the request currently being processed, and all subsequent requests in the task queue, are abnormally terminated with a ‘resource unavailable’ condition. This will generate an SQL error code of -652 and reason code of X'1015' to the calling application.

When a task is force purged, the current value of the MAXTRDR, MAXDRDR, MAXTWTR or MAXSCHED parameters is automatically decremented by 1, depending on which type of task has been terminated. Once purged, a task will not be available for object recall processing. Its status will appear as 'TERM' in any subsequent status display.

For busy tape reader and writer tasks, the current tape volume in use will be de-allocated. The tape will be rewound and dismounted, and the tape drive freed by DB2 Manager.

Tasks may only be force purged if a normal purge request has already been queued for that task.

Example:

```
F LCDDB2,D
```

```
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100
OTD20062 Readtimeout=0005 Writetimeout=0005
OTD20062 Maxtrdr=004 Maxdrdr=004 Maxtwtr=004 Maxsched=004 Maxqlen=100
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=BUSY Qlen=001
DBID=ST01 Stg_level=0 Tape=00002
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=WAIT Qlen=000
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=WAIT Qlen=000
OTD20063 Task D002: Status=WAIT Qlen=000
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=WAIT Qlen=000
OTD20063 Task W002: Status=WAIT Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
..........
F LCDDB2,P T=T1
OTD20073 Task T001 purge request is being processed
..........
F LCDDB2,F T=T1
OTD20060 Task T001 has been shutdown successfully
..........
F LCDDB2,D
OTD20062 Maxtrdr=003 Maxdrdr=004 Maxtwtr=004 Maxsched=004 Maxqlen=100
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100
OTD20062 Readtimeout=0005 Writetimeout=0005
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=TERM Qlen=000
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=WAIT Qlen=000
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=WAIT Qlen=000
OTD20063 Task D002: Status=WAIT Qlen=000
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=WAIT Qlen=000
OTD20063 Task W002: Status=WAIT Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
```

Purge task

```
P(URGE) T(ASK) = (T|D|W|S)nnn
```

The PURGE command is used to terminate a tape or disk reader task, a writer task or a scheduler task in the DB2 Manager control region. All outstanding requests in the task queue will be allowed to complete normally before task termination.

The command may be abbreviated to a single 'P'. The TASK parameter may be abbreviated to a single 'T'. The value 'nnn' must be numeric and should be preceded by the character ‘T’ for tape reader tasks, ‘D’ for disk reader tasks, ‘W’ for writer tasks and ‘S’ for scheduler tasks. If this character is omitted, a default value of ‘T’ will be used. The task identifier entered here must correspond to a valid reader task number, as displayed by the status summary command.

For waiting tasks (status = WAIT), the purge request will be executed immediately. The identified task will be terminated, and will no longer be available for object recall processing. It will appear with a type of TERM in subsequent status display requests. Any tape allocation held by a purged tape reader or writer task will be released - this will cause the allocated tape to be rewound and dismounted, and the tape drive to be freed by DB2 Manager.

For busy tasks (status=BUSY), a shutdown request will be added to the task queue. This will allow requests which were outstanding at the time the purge command was entered to be completed as normal. The task will then be terminated, and will no longer be available for processing. It will appear with a type of TERM in subsequent status display requests. For tape reader tasks, the tape being processed by the identified task will be released, whether it would have normally been retained on the tape drive after use or not.

Requests to purge a busy task for which a purge request has already been queued will be rejected.

When a task is purged, the current value of the MAXTRDR, MAXDRDR, MAXTWTR or MAXSCHED parameters is automatically decremented by 1, depending on the type of task being purged.

Example:

```
F LCDDB2,D
OTD20062 Maxtrdr=003 Maxdrdr=004 Maxtwtr=004 Maxsched=004 Maxqlen=100
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100
OTD20062 Readtimeout=0005 Writetimeout=0005
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=TERM Qlen=000
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=WAIT Qlen=000
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=WAIT Qlen=000
OTD20063 Task D002: Status=WAIT Qlen=000
```

```
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=WAIT Qlen=000
OTD20063 Task W002: Status=WAIT Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
......
F LCDDB2,P T=D1
OTD20073 Task D001 purge request is being processed
OTD25409 Disk rdr task 001 terminated
......
F LCDDB2,D
OTD20062 Maxtrdr=003 Maxdrdr=003 Maxtwtr=004 Maxsched=004 Maxqlen=100
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100
OTD20062 Readtimeout=0005 Writetimeout=0005
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=TERM Qlen=000
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=WAIT Qlen=000
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=TERM Qlen=000
OTD20063 Task D002: Status=WAIT Qlen=000
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=WAIT Qlen=000
OTD20063 Task W002: Status=WAIT Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
```

Alter DB2 Manager configuration

```
S(ET) { MAXTRDR = nnn
      { MAXDRDR = nnn
      { MAXTWTR = nnn
      { MAXSCHED = nnn
      { MAXQLEN = nnn
      { RETAINTAPE = nnnn
      { TAPEWAIT = nnnn
      { OBJSIZE = nnnnn
      { READTIMEOUT = nnnn
      { WRITETIMEOUT = nnnn
```

The SET command is used to alter the current values of the DB2 Manager MAXTRDR, MAXDRDR, MAXTWTR, MAXSCHED, MAXQLEN, RETAINTAPE, TAPEWAIT, OBJSIZE, READTIMEOUT, and

WRITETIMEOUT parameters. The command may be abbreviated to the single character 'S'.

The values of MAXTRDR, MAXDRDR, MAXSCHED, MAXQLEN, RETAINTAPE, TAPEWAIT, OBJSIZE, READTIMEOUT and WRITETIMEOUT are initially set during DB2 Manager control region initialization from the corresponding parameters in the TAPECNTL and ENVCNTL members of the DB2 Manager parameter library.

Valid parameters for the SET command are:

MAXTRDR - DB2 Manager will alter the current setting of the MAXTRDR parameter to the value specified in the SET command. The value ‘nnn’ must be a 1-3 digit numeric value in the range 0-255.

If the value of MAXTRDR is to be increased, DB2 Manager will increase the number of initialized tape reader tasks, by restarting terminated tasks, and/or adding new tasks.

If the value of MAXTRDR is to be decreased, DB2 Manager will terminate a number of existing tasks equal to the decrease in the value of MAXTRDR. These tasks will appear with the status TERM in any subsequent status display, and will not be available for object recall processing.

Note that processing of this command may take a little time for significant changes in the value of MAXTRDR.

MAXDRDR - DB2 Manager will alter the current setting of the MAXDRDR parameter to the value specified in the SET command. The value ‘nnn’ must be a 1-3 digit numeric value in the range 0-255.

If the value of MAXDRDR is to be increased, DB2 Manager will increase the number of initialized disk reader tasks, by restarting terminated tasks, and/or adding new tasks.

If the value of MAXDRDR is to be decreased, DB2 Manager will terminate a number of existing disk reader tasks equal to the decrease in the value of MAXDRDR. These tasks will appear with the status TERM in any subsequent status display, and will not be available for object recall processing.

Note that processing of this command may take a little time for significant changes in the value of MAXDRDR.

MAXTWTR - DB2 Manager will alter the current setting of the MAXTRDR parameter to the value specified in the SET command. The value ‘nnn’ must be a 1-3 digit numeric value in the range 1-255.

If the value of MAXTWTR is to be increased, DB2 Manager will increase the number of initialized writer tasks, by restarting terminated tasks, and/or adding new tasks.

If the value of MAXTWTR is to be decreased, DB2 Manager will terminate a number of existing writer tasks equal to the decrease in the value of MAXTWTR. These tasks will appear with the status TERM in any subsequent status display, and will not be available for object recall processing.

Note that processing of this command may take a little time for significant changes in the value of MAXTWTR.

MAXSCHED - DB2 Manager will alter the current setting of the MAXSCHED parameter to the value specified in the SET command. The value ‘nnn’ must be a 1-3 digit numeric value in the range 1-255.

If the value of MAXSCHED is to be increased, DB2 Manager will increase the number of initialized tape reader tasks, by restarting terminated tasks, and/or adding new tasks.

If the value of MAXSCHED is to be decreased, DB2 Manager will terminate a number of existing tasks equal to the decrease in the value of MAXSCHED. These tasks will appear with the status TERM in any subsequent status display, and will not be available for object recall processing.

Note that processing of this command may take a little time for significant changes in the value of MAXSCHED.

MAXQLEN - DB2 Manager will alter the current setting of the MAXQLEN parameter to the value specified in the SET command. ‘nnn’ must be a 1-3 digit numeric value in the range 1-256.

RETAINTAPE - DB2 Manager will alter the current setting of the RETAINTAPE parameter to the value specified in the SET command. ‘nnnn’ must be a 1-4 digit numeric value in the range 0-1440.This change will take immediate effect.

TAPEWAIT - DB2 Manager will alter the current setting of the TAPEWAIT parameter to the value specified in the SET command. ‘nnnn’ must be a 1-4 digit numeric value in the range 0-1440. This change will take immediate effect.

OBJSIZE - DB2 Manager will alter the current setting of the OBJSIZE parameter to the value specified in the SET command. ‘nnnnn’ must be a 1-5 digit numeric value in the range 1-

65535. This change will take effect the next time a database is connected for update processing by the DB2 Manager table migration utility.

READTIMEOUT - DB2 Manager will alter the current setting of the READTIMEOUT parameter to the value specified in the SET command. ‘nnnn’ must be a 1-4 digit numeric value in the range 1-1440. This change will take immediate effect.

WRITETIMEOUT - DB2 Manager will alter the current setting of the WRITETIMEOUT parameter to the value specified in the SET command. ‘nnnn’ must be a 1-4 digit numeric value in the range 1-1440. This change will take immediate effect.

Example:

```
F LCDDB2,D
OTD20062 Maxtrdr=003 Maxdrdr=003 Maxtwtr=004 Maxsched=004 Maxqlen=100
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100 Trace=OFF
OTD20062 Readtimeout=0005 Writetimeout=0005
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=TERM Qlen=000
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=WAIT Qlen=000
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=TERM Qlen=000
OTD20063 Task D002: Status=WAIT Qlen=000
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=WAIT Qlen=000
OTD20063 Task W002: Status=WAIT Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
......
F LCDDB2,SET MAXTWTR=2
OTD20052 DB2 Manager configuration altered successfully
OTD26009 Tape wtr task 001 terminated
OTD26009 Tape wtr task 002 terminated
......
F LCDDB2,SET READTIMEOUT=10
OTD20052 Configuration altered successfully
......
F LCDDB2,D
OTD20062 Maxtrdr=003 Maxdrdr=003 Maxtwtr=002 Maxsched=004 Maxqlen=100
OTD20062 Retaintape=0001 Tapewait=0004 Objsize=00100 Trace=OFF
OTD20062 Readtimeout=0010 Writetimeout=0005
OTD20062 Scheduler: Queue is unlocked; Queue length=000
OTD20063 Task T001: Status=TERM Qlen=000
OTD20063 Task T002: Status=WAIT Qlen=000
OTD20063 Task T003: Status=WAIT Qlen=000
OTD20063 Task T004: Status=WAIT Qlen=000
OTD20063 Task D001: Status=TERM Qlen=000
```

```
OTD20063 Task D002: Status=WAIT Qlen=000
OTD20063 Task D003: Status=WAIT Qlen=000
OTD20063 Task D004: Status=WAIT Qlen=000
OTD20063 Task W001: Status=TERM Qlen=000
OTD20063 Task W002: Status=TERM Qlen=000
OTD20063 Task W003: Status=WAIT Qlen=000
OTD20063 Task W004: Status=WAIT Qlen=000
OTD20063 Task S001: Status=WAIT Qlen=000
OTD20063 Task S002: Status=WAIT Qlen=000
OTD20063 Task S003: Status=WAIT Qlen=000
OTD20063 Task S004: Status=WAIT Qlen=000
OTD20058 Display processing completed
```

Terminate DB2 Manager control region

```
SHUTDOWN (IMMED)
```

The SHUTDOWN command is used to terminate the DB2 Manager control region. The optional IMMED parameter is used to control the shutdown procedure.

If IMMED is not specified, DB2 Manager will allow all outstanding requests to be completed normally before terminating the control region. However, no new requests will be serviced by the control region after shutdown has been requested.

If IMMED is specified, DB2 Manager will perform an immediate shutdown; all current and outstanding retrieval requests will be terminated with a ‘resource unavailable’ condition (SQL –652 error with reasons code X’1015’) and the DB2 Manager control region terminated.

Terminating the control region will disable support for retrieval of migrated rows via DB2 Manager. The shutdown request should not therefore be performed without first verifying that no access to migrated rows will be made while the control region is inactive.

If COMMAND=REPLY has been specified (or defaulted) in the TAPECNTL parameter library member, DB2 Manager will issue message OTD20092 on receiving the SHUTDOWN command:

```
OTD20092 Do you want to proceed with shutdown (y/n)?
```

To continue with shutdown processing, reply 'Y'. Any other reply will abort shutdown processing. DB2 Manager will redisplay the standard operator reply message OTD20000, and control region processing will continue as normal.

If COMMAND=MODIFY has been specified in the TAPECNTL member, DB2 Manager will not issue a confirmation request. Shutdown processing will continue automatically.

Example:

```
13 OTD20000 Enter request:
R 13,SHUTDOWN
```

```
IEE600I REPLY TO 13 IS;SHUTDOWN
14 OTD20092 Do you want to proceed with shutdown (y/n)?
R 14,Y
IEE600I REPLY TO 14 IS;Y
OTD20093 Shutdown in progress
OTD20060 Task H001 has been shutdown successfully
OTD25009 Tape rdr task 004 terminated
OTD20060 Task T004 has been shutdown successfully
OTD25009 Tape rdr task 003 terminated
OTD20060 Task T003 has been shutdown successfully
OTD25009 Tape rdr task 002 terminated
OTD20060 Task T002 has been shutdown successfully
OTD25009 Tape rdr task 001 terminated
OTD20060 Task T001 has been shutdown successfully
OTD25409 Disk rdr task 004 terminated
OTD20060 Task D004 has been shutdown successfully
OTD25409 Disk rdr task 003 terminated
OTD20060 Task D003 has been shutdown successfully
OTD25409 Disk rdr task 002 terminated
OTD20060 Task D002 has been shutdown successfully
OTD25409 Disk rdr task 001 terminated
OTD20060 Task D001 has been shutdown successfully
OTD26009 Tape wtr task 004 terminated
OTD20060 Task W004 has been shutdown successfully
OTD26009 Tape wtr task 003 terminated
OTD20060 Task W003 has been shutdown successfully
OTD26009 Tape wtr task 002 terminated
OTD20060 Task W002 has been shutdown successfully
OTD26009 Tape wtr task 001 terminated
OTD20060 Task W001 has been shutdown successfully
OTD22009 Scheduler task 004 terminated
OTD20060 Task S004 has been shutdown successfully
OTD22009 Scheduler task 003 terminated
OTD20060 Task S003 has been shutdown successfully
OTD22009 Scheduler task 002 terminated
OTD20060 Task S002 has been shutdown successfully
OTD22009 Scheduler task 001 terminated
OTD20060 Task S001 has been shutdown successfully
OTD20095 Communications table storage area freed
OTD20096 Shutdown processing complete
```

■ SMF processing

DB2 Manager provides an option to implement the recording of migrated row storage retrieval statistics by the control region, using the IBM System Management Facility (SMF).

This option is invoked by users via the optional SMFRECID parameter of the ENVCNTL member of the DB2 Manager parameter library.

Omitting this parameter, or specifying an SMFRECID of zero, will suppress creation of SMF records by DB2 Manager.

Specifying a non-zero value, in the range 128 to 255, will cause DB2 Manager to create SMF records with the specified numeric identifier.

If requested, DB2 Manager will create SMF records at the following points:

a. When all requests for retrieval of rows from a single tape cartridge volume have been completed by a tape reader task (ie) immediately before a busy task returns to wait status. The SMF record will contain one request section for each request processed during invocation of that reader task.

b. When the limit of 100 request sections has been created by a tape reader task while processing a request queue. An SMF record will be written by the reader task before continuing to process the request queue. One or more additional SMF records will subsequently be written by the reader task, either when the limit of 100 request sections has again been reached, or after all requests in the queue have been processed.

   Note that there is no direct relationship between the number of requests that may be processed during a single invocation of a tape reader task, and the setting of the MAXQLEN parameter. After processing a request, the queue length will be reduced by one; however, new requests may continue to arrive and be added to the queue while the driver task is busy. New requests will only be rejected if the number of outstanding requests has reached the value of MAXQLEN.

   In this way, there is no limit to the number of requests which may be processed during a single invocation of a reader task, and consequently no limit to the number of SMF records which may be written during that invocation.

c. When all queued requests for retrieval from the same Archive Manager database have been completed by a disk reader task (ie) immediately before a busy disk reader task processes a request for retrieval from a different Archive Manager database, or returns to wait status if there are no more requests in the queue.

d. During row migration processing, when the limit of 100 request sections has been created by a writer task. An SMF record will be written by the writer task before continuing with row migration processing. One or more additional SMF records will subsequently be written by the writer task, either when the limit of 100 request sections has again been reached, when a disconnect or commit request is received, or when the task has timed out waiting for receipt of another store request for the Archive Manager database being updated.

Each request section in an SMF record will contain four timestamps relating to the DB2 Manager retrieval request described by the record. These timestamps are issued at the following points:

1. When the request for access to a migrated row is intercepted by DB2 Manager.

2. Start of request processing by the control region scheduler.

3. Start of request processing by the reader or writer task in the control region.

4. Termination of request processing by the reader or writer task in the control region.

Analysis of timestamp intervals will provide timings for the following activities:

4 - 1 :	total time for DB2 Manager to process the request.
2 - 1:	time spent in control region scheduler queue.
3 - 2:	time spent in reader or writer task queue.
4 - 3:	time taken to retrieve object from tape or disk and locate migrated row (reader tasks) or to add migrated row to Archive Manager object (writer tasks).

Careful analysis of this information will assist in identifying possible performance problems during object retrieval processing.

The following sections describe the individual components that constitute the DB2 Manager SMF record.

SMF header section

Each SMF record will have a standard SMF header section of length 18 bytes.

Offset	*Length*	*Format*	*Description*
0	2	bin	Record length. This is the length of the entire SMF record, including this field.
2	2	bin	Segment descriptor - binary zeroes.
4	1	bin	System indicator - binary zeroes.
5	1	bin	Record type (equal to the value of the SMF record identifier specified defined for the database being processed).
6	4	bin	Time since midnight, in hundredths of a second, that the SMF record was written.
10	4	packed	Date record was written, in the format 0cyydddF, where F is the sign.
14	4	char	System identification (from CVT).

Record descriptor section

The SMF header section is immediately followed by the DB2 Manager record descriptor section. It has a fixed length of 38 bytes.

Offset	*Length*	*Format*	*Description*
18	6	bin	Reserved.
24	4	bin	Offset of database section. This is the offset from the start of the SMF record of the database section of the record.
28	2	bin	Length of database section. This is the length in bytes of a single database section.
30	2	bin	The number of database sections present in the record (always = 1).
32	4	bin	Offset of request section. This is the offset from the start of the SMF record of the first request section in the record.
36	2	bin	Length of request section. This is the length in bytes of a single section. This will always be 96 bytes.
38	2	bin	The number of request sections present in the record.
40	16	-	Reserved for future use.

Database section

The storage group section starts at the offset specified in the record descriptor section of the DB2 Manager SMF record. It gives information about the database being processed for object storage or retrieval. It has a fixed length of 24 bytes. Only one database group section will be present per SMF record.

Offset	*Length*	*Format*	*Description*
0	1	Char	DB2 Manager task issuing request: ’T’ – tape reader task ’D’ – disk reader task ’W’ – tape writer task

Offset	*Length*	*Format*	*Description*
1	3	Char	DB2 Manager reader/writer task number.
4	4	char	Identifier of Archive Manager database being processed.
8	16	-	Reserved for future use

Request section

The request section starts at the offset specified in the record descriptor section of the DB2 Manager SMF record. It gives information about each of the individual requests processed by the DB2 Manager reader or writer task. It has a fixed length of 96 bytes. A maximum of 100 request sections may be present per SMF record.

Offset	***Length***	***Format***	***Description***
0	8	bin	Request start timestamp. This gives the date and time that the request was received by the DB2 Manager edit routine. It has the format: bytes 0-3: time in hundredths of a second since midnight that request processing started. bytes 4-7: date that request processing started, in packed decimal format 0cyydddF, where F is the sign.
8	8	bin	Scheduler timestamp. This gives the date and time that the request was received by the DB2 Manager control region scheduler. It has the format: bytes 0-3: time in hundredths of a second since midnight that the request was intercepted. bytes 4-7: date that the request was intercepted, in packed decimal format 0cyydddF, where F is the sign.
16	8	bin	Reader/writer start timestamp. This gives the date and time that the request was first processed by the DB2 Manager reader or writer task. It has the format: bytes 0-3: time in hundredths of a second since midnight that the request was first processed. bytes 4-7: date that the request was first processed, in packed decimal format 0cyydddF, where F is the sign.
24	8	bin	Request end timestamp. This gives the date and time that processing for this request was completed by the DB2 Manager control region. It has the following format: bytes 0-3: time in hundredths of a second since midnight that

Offset	***Length***	***Format***	***Description***
			request processing ended. bytes 4-7: date that request processing ended, in packed decimal format 0cyydddF, where F is the sign.
32	2	char	Return code.
34	2	bin	Supplementary return code 1
36	2	bin	Supplementary return code 2
38	2	Bin	Request type: 0 – read request 1 – connect request 2 – store request 3 – commit request 4 – disconnect request 5 – disconnect after timeout
40	1	Char	Storage level (0-7)
41	1	Char	Object residency indicator (‘T’ = tape, ‘D’ = disk; reader tasks only – contains a space for writer tasks).
42	2	bin	Sequence number within database of tape or disk dataset being processed (reader tasks only – contains low values for writer tasks).
44	4	bin	*Tape resident objects*: identifier of tape or block of object containing retrieved row (reader tasks only – contains low values for writer tasks). *Disk resident objects*: byte 0: hex id of disk segment containing object bytes 1-3: id of disk block containing object (in form TTTTRR)
48	4	Bin	Primary key of object containing stored/retrieved row.
52	8	char	Archive date of object containing stored/retrieved row (in format YYYYMMDD).

Offset	*Length*	*Format*	*Description*
60	2	Bin	Record number within object containing stored/retrieved row
62	2	Bin	Number of records containing stored/retrieved row
64	8	char	Requester’s jobname
72	8	char	Requester’s userid.
80	4	char	Identifier of system on which request originated
84	12	-	Reserved

■ Operational considerations

This section discusses miscellaneous topics relating to the operation of the DB2 Manager control region and its environment. This includes a discussion of performance considerations for retrieval of migrated rows from an Archive Manager database using DB2 Manager.

Use of the MAXTRDR parameter

The MAXTRDR parameter is used to limit the number of tape cartridge drives that DB2 Manager will allocate concurrently for performing simultaneous access to multiple tape volumes in response to migrated row retrieval requests. It is intended to enable users to prevent DB2 Manager from depleting the number of available cartridge drives on a system to the extent that other work will be unable to complete.

Once at the MAXTRDR limit, DB2 Manager will reject any other request which would require an extra tape drive in order for it to be satisfied, with a ‘resource unavailable’ condition, or optionally queue the request internally if a non-zero value has been specified for the TAPEWAIT parameter in the TAPECNTL parameter library member. In this latter case DB2 Manager will retry the request at 1-second intervals until a tape drive becomes available or until the TAPEWAIT interval is exceeded.

Rejection of a request due to the MAXTRDR value having been reached or due to the TAPEWAIT interval having been exceeded will generate an SQL –652 error and reason code (in field SQLERRD(6) in the SQLCA) of X’1015’. The value chosen for MAXTRDR should ensure that occurrence of this condition is minimized, while still allowing other work on the system to continue without delay. If necessary, the value of MAXTRDR may be varied during the day to correspond to variations in the system workload.

In a mixed operating environment which consists of both library and free-standing tape drives, the STK Host Software Component will ensure that all primary DB2 Manager tapes will be located in a library storage module. The value of MAXTRDR used in this environment should not exceed the total number of drives available in the library configuration, otherwise allocation requests for DB2 Manager tapes may be made to free-standing drives. This will introduce unpredictable delays into the migrate row retrieval process; this is obviously unacceptable in an online processing environment.

In any case, the value of MAXTRDR should not exceed the overall total number of drives available, as this will cause a significant increase in the incidence of allocation recovery, and consequent rejection of retrieval requests.

Use of the MAXQLEN parameter

The MAXQLEN parameter places a limit on the maximum number of requests which may be queued for a tape reader task (i.e.) the maximum number of outstanding requests that may exist for retrieval of objects from any one tape volume. It is intended to allow users to place a limit on the time that any single request will spend in a tape reader task request queue.

Requests are added to the task request queue as they arrive from other DB2 Manager components (in response to application SQL commands) and removed from the queue as they are completed. The request queue may thus be in a state of change while it is being processed. The MAXQLEN parameter places a limit on the number of outstanding requests in the queue, not the total number of requests which may be processed from the queue by the tape reader task.

If a tape needs to be mounted in response to the first request for that tape volume that is received by a reader task, it will take the time required to mount the tape and to locate the object containing the migrated row, in order to process that request. Subsequent requests in the same queue will only require an object locate operation.

If multiple requests for retrieval from the same tape volume arrive within a very short period of time, then the last request to arrive will have to wait for all preceding requests to be processed.

The above information may be used to select an appropriate value of the MAXQLEN parameter at your installation, in order to limit the length of time that a request must wait in the tape reader task queue when multiple simultaneous requests have been issued.

Once at the MAXQLEN limit, DB2 Manager will reject any other request which would cause this limit to be exceeded, with a ‘resource unavailable’ condition, or optionally queue the request internally (if a non-zero value has been specified for the TAPEWAIT parameter in the TAPECNTL parameter library member). In this latter case DB2 Manager will retry the request until a tape drive becomes available or until the TAPEWAIT interval is exceeded.

Rejection of a request due to the MAXQLEN value having been reached or due to the TAPEWAIT interval having been exceeded will generate an SQL –652 error with a reason code of X'1015' in field SQLERRD(6) in the SQLCA.

Use of the MAXDRDR parameter

The MAXDRDR parameter is used to control the number of disk reader tasks which are active in the DB2 Manager control region. A disk reader task is used to perform retrieval of a migrated row from an Archive Manager database when a disk (‘K’) copy exists of the object containing the row.

If the value of MAXDRDR is greater than 1, DB2 Manager will assign a new disk object retrieval request to the disk reader task with the shortest request queue. In this way, the length of the request queues will be balanced across the allocated disk reader tasks. Increasing the value of MAXDRDR will decrease the average disk reader task request queue length. However, there will be a corresponding increase in the amount of virtual storage required by the DB2 Manager control region. For most operational environments, the default MAXDRDR value of 4 will be sufficient.

If the value of MAXDRDR is 0, then all object retrieval requests, whether objects have a disk copy or not, will be processed by the tape reader tasks currently allocated in the control region. Each tape reader task will issue a standard Archive Manager object retrieval request; if a disk copy of an object exists then Archive Manager will satisfy the retrieval request from disk (i.e. the object will still be retrieved from disk, even though it is being processed by a tape reader task). Setting the MAXDRDR value to 0 is not recommended in these circumstances, as it may cause disk object retrievals to be rejected unnecessarily if there are no available tape reader tasks to process the request.

If an error occurs when Archive Manager attempts to retrieve the disk copy of an object, Archive Manager will automatically re-drive the retrieval attempt from tape. In these circumstances it is possible that a disk reader task will attempt to process a tape dataset. This may cause contention for tape drives, resulting in failure of the retrieval request, if all available drives are currently being processed by the tape reader tasks. Retrieval of the tape copy of an object by a disk reader task will not impact the ability of that task to continue accepting and processing further disk object retrieval requests.

Allocation recovery

Allocation recovery occurs on MVS systems when there are no tape drives available to satisfy an allocation request. In this case, the following message will be output by the system:

```
IEF238D jjj - REPLY DEVICE NAME, 'WAIT' OR 'CANCEL'
```

Normal practice is for an operator (or automated operator facility) to reply WAIT to this request. The job requiring the allocation will then wait until a drive becomes available before being able to continue.

In the DB2 Manager environment, lack of available tape drives will cause the DB2 Manager control region to enter allocation recovery. Issuing a reply of WAIT will now mean that the application issuing the SQL command which caused the tape allocation request will have to wait indefinitely for another job on the system to release a drive, before continuing.

In this case, the end-user response time would be unpredictable. For this reason, it is recommended that a reply of 'CANCEL' be made to the IEF238D message. This will be interpreted by DB2 Manager as a 'drive unavailable' condition, and the retrieval request will be rejected with a ‘resource unavailable’ condition or optionally queued internally (if a non-zero value has been specified for the TAPEWAIT parameter in the TAPECNTL parameter library member). In this latter case DB2 Manager will retry the request until a tape drive becomes available or until the TAPEWAIT interval is exceeded. .

It is recommended that the reply of 'CANCEL' be performed by an automated operator reply facility, if present on the system, rather than via manual operator entry. In this way, there will be no delay between DB2 Manager receiving the retrieval request, and the rejection of the request with a ‘resource unavailable’ condition. This will minimize delays to users in receiving notification of the unavailability of resources in performing the object access.

Shutdown processing

As an immediate shutdown does not allow outstanding migrated row storage and retrieval requests to complete, whenever possible the DB2 Manager control region should be shutdown normally (ie) use the SHUTDOWN operator command without the IMMED parameter.

In this context, it is recommended that operators issue a summary status display command before issuing the DB2 Manager shutdown request.

If there are any outstanding requests identified by the summary status display, these should be allowed to complete before issuing a normal DB2 Manager shutdown request. If necessary, an immediate shutdown request may be issued in order to abnormally terminate outstanding requests, and perform DB2 Manager shutdown processing.

Additionally, if the scheduler lock, as displayed from the summary status request, is constantly unavailable (ie. permanent LOCKED status), then an immediate shutdown should be issued, as a normal shutdown attempts to acquire this lock before proceeding. This situation, if it occurs, indicates an internal DB2 Manager error condition, and should be communicated to your DB2 Manager product support representative.

In extreme cases, the DB2 Manager control region may be cancelled if required. Canceling the control region has the following consequences:

- Any outstanding requests will be left unsatisfied in a wait state (ie) control will never be returned to the calling application. These applications will have to be abnormally terminated by an external source.

- Extended CSA storage used by DB2 Manager (approximately 212k in extended subpool 241) will not be released, and will be unavailable for use by other jobs in the system until the next IPL.

'Resource unavailable' condition

DB2 Manager will return a ‘resource unavailable’ condition to an SQL command which generates access to a migrated row (SQL –652 error and reason code of X'1015' set in field SQLERRD(6) in the SQLCA) in the following circumstances:

1. a row retrieval request requires a new tape cartridge volume to be mounted, but DB2 Manager is unable to allocate a new drive because the MAXTRDR limit has already been reached.
2. a row retrieval request requires access to a tape cartridge volume that is already being processed, but the number of requests already queued for that volume is at the limit specified by the MAXQLEN parameter.
3. allocation recovery has occurred due to lack of available tape cartridge drives, and a reply of 'CANCEL' was made to the IEF238D message.
4. a ‘force purge’ command was issued by the operator for the task currently processing the queue containing the row retrieval request.
5. an immediate DB2 Manager control region shutdown was issued by the operator while a row retrieval request was being processed, or while it was in a task request queue.
6. the DB2 Manager control region is in the process of terminating when the SQL command was issued.

Note that the TAPEWAIT parameter of the TAPECNTL parameter library member may be used to internally queue requests which have been rejected in circumstances 1 to 3 above. In this case a request will be retried until sufficient resources become available to satisfy it, or until the TAPEWAIT interval has been exceeded, at which time it will be rejected as above. Use of this facility may reduce the likelihood of ‘resource unavailable’ conditions occurring.

This page is intentionally left blank

Table migration processing

5

■ Introduction

DB2 Manager provides functions to perform the migration of rows from DB2 to Archive Manager storage. These functions are provided by the DB2 Manager table migration utility, OTDBP100.

The table migration utility is executed as a batch process, and should be scheduled to run at regular intervals in order to migrate rows from DB2 to an Archive Manager database. The frequency of execution of this procedure will be controlled by operational factors, such as the overall amount of DB2 disk storage available, the rate of insertion of new rows, the frequency of update of migrated rows, the rate of retrieval of migrated rows etc.

Careful planning is required when implementing a table migration strategy, in order to optimize system performance by migrating rows at the appropriate stage of their existence. Migration should be performed so that DB2 disk storage requirements may be minimized without introducing performance problems through the creation of an Archive Manager tape recall activity rate which exceeds the performance capacity of the installed tape library configuration.

DB2 Manager requires that a table from which rows are to be migrated is enabled for this type of processing. “Enabling a DB2 table for migration processing” on page 75 describes the activities required to enable a table for migration processing.

Following successful completion of migration processing, some DB2 database management is needed if it is required to reclaim disk space released through the migration of rows to Archive Manager storage. “DB2 database reorganization” on page 88 explains the issues relating to this requirement and describes how to do this.

■ Enabling a DB2 table for migration processing

Before initiating DB2 Manager row migration processing for a DB2 table, the table must be enabled for this type of processing. This consists of a number of activities as described in this section. These activities must be successfully completed for each DB2 table which is to be processed by DB2 Manager.

Restrictions

DB2 tables which are to be eligible for processing by DB2 Manager must adhere to the following restrictions:

- Each row in the table must be indexed by a unique key. The indexing key can consist of one or multiple columns in the table.
- The table must not contain large objects (LOBs), ROWID or identity columns or “distinct types”.

A table which does not satisfy all the above criteria will not be processed by the DB2 Manager table migration utility.

Enabling activities

In order to enable a DB2 table for row migration by DB2 Manager, the following actions must be performed:

- A new partitioned tablespace must be created to hold a copy of the table being enabled for migration processing. This should have one more partition than the number of partitions in the tablespace containing the table to be enabled for migration. The last partition in this tablespace will be used for storage of all archive row stubs. Each stub will be 18 bytes in length, and there will be one stub for each row in the table which has been migrated to Archive Manager. This partition should be large enough to hold archive stubs for the maximum number of migrated rows present in the database.
- A new table must be created in this tablespace. This should be an exact copy of the original table, with the following modifications:
    - The statement ‘EDITPROC OTDBP300’ is added to the SQL CREATE TABLE command.
    - An additional column ‘OTDBIND’ is added to the SQL CREATE TABLE command, with the following specification:

      OTDBIND CHAR(1) DEFAULT ‘0’

Example:

```
CREATE TABLE OTDBSYST.TAB1A
       (
        FIELD1   CHAR(1)              ,
        FIELD2   CHAR(255)            ,
        FIELD3   VARCHAR(255)         ,
        FIELD4   GRAPHIC(1)           ,
        FIELD5   VARGRAPHIC(127)      ,
        FIELD6   SMALLINT     NOT NULL,
        FIELD7   INTEGER      NOT NULL,
        FIELD8   DECIMAL(7,0) NOT NULL,
        FIELD9   REAL                 ,
        FIELD10  DOUBLE               ,
        FIELD11  TIME                 ,
        FIELD12  DATE         NOT NULL,
 FIELD13  TIMESTAMP            ,
 OTDBIND  CHAR(1)   DEFAULT '0'
)
IN OTDBSYST.TAB1TS
EDITPROC OTDBP300;
```

- For existing non-partitioned tables, a new partitioning index to the new table must be created, as follows:

```
CREATE INDEX dbname.indexname
 ON dbname.tabname
    (
     OTDBIND  ASC
    )
    CLUSTER
    (PART 1 VALUES('0'),
     PART 2 VALUES('1'))
        ....
```

- For existing partitioned tables, the definition of the existing partitioning index should be copied with the following modifications:

```
CREATE INDEX dbname.indexname
       ON dbname.tabname
       (
        OTDBIND  ASC,
        field1 ASC/DESC,
        ........,
        fieldn ASC/DESC
        )
        CLUSTER
        (PART 1 VALUES(‘0’,a,...,b)
         PART n VALUES (‘0’,c,...,d)
         PART n+1 VALUES (‘1’,y,...,z))    ...
```

where:

field1 is the first column in the existing clustering index

fieldn is the last column in the existing clustering index

n is the number of partitions defined in the existing clustering index

‘a’ ... ‘b’ are the values of columns field1 to fieldn used to define partition 1 in the existing clustering index

‘c’ ... ‘d’ are the value of columns field1 to fieldn used to define partition n in the existing clustering index

‘y’ ... ‘z’ are the lowest possible values for columns field1 to fieldn

- All other existing indexes should be copied without modification to index the new table.

- The contents of the existing table should be unloaded, and loaded into the new table. This will create an exact copy of the original table, with the modifications described above.
- The original table should be dropped from DB2.
- A view should be created of the new table. The view must have the same name as that of the original table, and should be defined as follows:

  ```
  CREATE VIEW dbname.tabname1
  AS SELECT
  field1,
  field2,
  ...
  fieldn
  FROM
  dbname.tabname2
  ```

  where:

  tabname1 is the name of the existing table

  tabname2 is the name of the new table

  field1 is the first column in the existing table

  fieldn is the last column in the existing table
- All views on the existing table should be re-created as originally defined.

Following completion of migrated table enablement processing, a table will be eligible for processing by the DB2 Manager table migration utility.

No other implementation requirements are needed in order to enable migration and retrieval processing for a DB2 table. In particular, no modifications to existing application programs issuing SQL commands will be required. DB2 Manager will automatically enable retrieval, deletion and update of rows which have been migrated to an Archive Manager database.

Disabling activities

If it is required to disable DB2 Manager support for a DB2 table which had previously been enabled for row migration by DB2 Manager, the following actions must be performed:

- **Before performing any activities to disable DB2 Manager support for a DB2 table, all migrated rows must be recalled to the table using the DB2 Manager row retrieval utility OTDBP120**. Failure to perform this step prior to continuing with disabling processing will mean that data in migrated rows will no longer be accessible and may be permanently lost. Any attempt to access these rows from an SQL application will result in an SQL error.
- Views created for the migration-enabled table should be dropped.
- A new partitioned tablespace must be created to hold a copy of the table being disabled for migration processing. This should have one less partition than the number of partitions in the tablespace containing the

migration-enabled table – the last partition in this table should be removed when creating the disabled table. Alternatively, if the original tablespace used to hold the table prior to enabling it for migration processing still exists, this may be used.

- A new table must be created in this tablespace. If the original definition used to create the table prior to enabling it for migration processing exists, this should be used. Otherwise the definition should be an exact copy of that for the migration-enabled table, with the following modifications:
  - The name of the table should be the name given to the view created for accessing the migration-enabled table during migration enabling processing.
  - Remove the statement 'EDITPROC OTDBP300' from the SQL CREATE TABLE command.
  - Remove the column 'OTDBIND' from the SQL CREATE TABLE command.
- If the migration-disabled table is partitioned, modify the existing partitioning index by removing the column OTDBIND from the index definition, modifying the VALUES statement to remove the OTDBIND entries, and removing the PART entry for the last (removed) partition.

  If the migration-disabled table is not partitioned, remove the partitioning (PART) statements from the index definition.
- All other existing indexes should be copied without modification to index the migration-disabled table.
- The contents of the migration-enabled table should be unloaded, and loaded into the new table. Remove the entry for field OTDBIND from the SYSIN file used for the re-load job. This process will create an exact copy of the original table, with the modifications described above.
- All views on the existing table (other than that having the name of the migration-disabled table itself) should be re-created as originally defined.

A table will be disabled for migration processing following completion of the above activities. DB2 Manager will then no longer be required when accessing the table.

■ OTDBP100 - the table migration utility

Migration of rows from a DB2 table to Archive Manager storage is performed using the DB2 Manager table migration utility. This is executed as a batch job, and processes a single DB2 table per execution. An SQLIN dataset entry is used to specify the selection criteria for migration of table rows. This is specified in the form of an SQL SELECT command. Refer to “SQLIN entry specification“ on page 85 for a description of the specification and usage of this entry.

Rows from a single DB2 table will be stored in a single Archive Manager database (identified to the utility via the SQLIN file). In addition, an Archive Manager database may only contain rows from a single DB2 table. DB2 Manager enforces this one-for-one relationship between DB2 table and Archive Manager database during execution of the table migration utility.

Multiple table migration jobs can be executed concurrently. However, each job must process a different DB2 table – only one table migration procedure per DB2 table is permitted at any one time. The table migration utility communicates directly with the DB2 Manager control region in order to connect and disconnect the Archive Manager database being updated. Note that the DB2 Manager control region must be fully initialized prior to the start of table migration processing.

A control region writer task will be assigned to the procedure during this process. This task will store migrated rows in Archive Manager objects on the latest tape dataset in storage level 0 of the Archive Manager database. No other writer task may connect to the Archive Manager database for update processing while this is in progress.

In addition, tape reader tasks will not be able to satisfy retrieval requests for migrated rows stored in objects on the tape volume being updated by the writer task. A ‘resource unavailable’ error (SQL –652 error and a reason code of X’1015’ in field SQLERRD(6) in the SQLCA) will be returned to an application program which generates such a request. If access to such rows is required, it is recommended that Archive Manager (‘K’) copy processing be enabled in order to hold copies of objects from at least the latest tape volume in the database storage level on disk. This will permit retrieval of these objects using control region disk reader tasks. Alternatively, the Archive Manager “dynamic load balancing” facility could be implemented in order to satisfy retrievals from the appropriate backup volume, if available. Refer to the Archive Manager User Manual for a description of both these facilities.

Note that retrieval of migrated rows from objects held on tape volumes other than the latest in the database storage level will be able to continue without interruption while table migration processing is in progress for that Archive Manager database.

Functions

The table migration utility performs the following functions:

- It processes and validates a series of parameters from a SYSIN file, and an object selection parameter from file SQLIN which will govern execution of the utility. A parameter specification and validation report is written to the SYSPRINT dataset.
- It sequentially processes all unmigrated rows in the DB2 table identified in the SQLIN parameters and selects all rows which satisfy the migration criteria.
- Each selected row is passed to a writer task in the DB2 Manager control region for storage as a component of an object in the Archive Manager database specified in the SYSIN parameter. The number of rows per object is controlled via the OBJSIZE parameter in the ENVCNTL member of the DB2 Manager parameter library.
- On successful storage in the Archive Manager database, the migrated row is replaced by an 18-byte "archive stub", identifying that the row has been migrated and recording its location in the Archive Manager database. Index entries for the migrated row will be unaltered, and will continue to exist as they had prior to row migration. The archive stub is stored in the table's archive partition (created during enabling of the table for migration processing).
- The utility will commit updates to the Archive Manager database and the DB2 table at the frequency specified in the COMMIT execution parameter. Following successful completion of commit processing, access to all rows migrated since the last commit point (or since the start of utility processing for the first commit request) will be made via the DB2 Manager control region.
- When all rows have been successfully migrated, the utility will produce a processing report indicating successful completion and providing totals of the number of rows processed and the number of rows migrated during utility execution.
- If the utility encounters a serious error which prevents continued processing, it will terminate with a condition code 12 after producing an error report identifying the cause of the error and a processing report indicating unsuccessful completion and giving totals of the number of rows processed and the number of rows successfully migrated up to the last successful commit point.

JCL requirements

The following JCL may be used to execute the utility:

```
//*
//STEP100  EXEC PGM=OTDBP100,REGION=4096K
//*
//STEPLIB   DD  DSN=Database.Manager.load.library,DISP=SHR
//          DD  DSN=Archive.Manager.load.library,DISP=SHR
//*
//SYSIN     DD  DSN=Execution.parameter.file,DISP=SHR
//SQLIN     DD  DSN=Migration.criteria.file,DISP=SHR
//SYSPRINT  DD  SYSOUT=*
//*
```

DD entries should be specified as follows:

STEPLIB: Enter the names of the DB2 Manager and Archive Manager product load libraries.

SYSPRINT: This entry is used for all print reports created during utility execution.

SYSIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 byte records, and is used for specification of the parameters used to control utility execution. Refer to “SYSIN parameter specification” on page 82 for a description of how to specify parameter entries in this file.

SQLIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 bytes records, and is used for specification of the migration selection criteria to be used by the utility for selection of rows for migration from DB2 to DB2 Manager. Refer to “SQLIN entry specification” on page 85 for a description of how to specify a migration selection criteria entry in this file.

SYSIN parameter specification

Parameters used to control execution of the utility are specified in file SYSIN. This must be a sequential dataset or in-stream JCL file containing one or more fixed-length 80-byte records, each record containing one or more separate parameter entries. Each parameter entry must be completely contained within one record (ie) continuation of any single parameter entry from one record to another is not permitted. A single record may contain multiple parameter entries, each being separated by one or more spaces, and optionally a single comma (‘,’).

A parameter entry has the general format:

```
keyword = value
```

Each keyword and its associated value are separated by an '=' sign, and optionally one or more spaces. The first keyword parameter in a record may begin in any character position from 1 onwards. Parameters may extend up to character position 71 of the record. Individual keyword parameters may not be continued over more than one record.

An asterisk ('*') in character position 1 will indicate a comment record, which will be ignored for parameter validation processing.

The following parameters may be specified:

- LCDDBID = xxxx

  This is a mandatory parameter and specifies the 4-character identifier of the Archive Manager database to which rows are to be archived during execution of the utility. This database must have been identified in the JCL used to start the DB2 Manager control region (see "Control region initialization" on page 44). This value is used to identify the Archive Manager database in all DB2 Manager components.

  The utility will verify that the Archive Manager database contains only migrated rows from the DB2 table being processed in its current execution. If this is not the case, an error report is produced and the utility will terminate with condition code 12.

- COMMIT = nnnnn

  This is an optional parameter used to specify the frequency at which Archive Manager and DB2 commit processing is to be performed by the utility. 'nnnn' is a 1-5 digit numeric value in the range 1-65535, specifying the number of rows to be processed between each commit request. If this parameter is omitted, a default value of 1000 is used.

  Once commit processing has been successfully completed, all rows migrated since the last commit point (or since the beginning of the job, for the first commit call) will have been successfully stored in the Archive Manager database, and all subsequent access to those rows will be performed by DB2 Manager.

  The value specified (or defaulted) for the COMMIT parameter will also be used in conjunction with the global OBJSIZE parameter (specified in the ENVCNTL parameter library member) to determine the number of rows to be stored per object in the Archive Manager database. The actual number of rows per objects will be the lower of the two values (COMMIT or OBJSIZE).

- RESTART=YES|<u>NO</u>

  This is an optional parameter used to control restart processing. If the previous execution of the utility terminated abnormally, the parameter should be set to "YES". If omitted, the default value of "NO" is used.

  Failure to specify this parameter correctly may cause execution of the utility to fail with an Archive Manager connect database return code of '15', indicating that the utility is attempting normal connection to a database which is in an "uncommitted" condition.

- NEWTAPE=YES|<u>NO</u>

  This is an optional parameter used to cause a new tape volume in the Archive Manager database to be opened at the start of migration processing.   If omitted, the default value of “NO” is used.

- DB2SSID=xxxx

  This mandatory parameter is used to specify the subsystem identifier of the DB2 subsystem to be used during execution of the utility.   The subsystem identifier ‘xxxx’ can be 1-4 characters in length.

- PLAN=xxxxxxxx

  This is an optional parameter which is used to specify the DB2 plan name to be used during execution of the utility, where ‘xxxxxxxx’ is the 1-8 character identifier of the plan. If omitted, a default value of OTDBL100 is used. Refer to “Perform DB2 system modifications” page 25 for information about DB2 Manager plan creation processing.

- S100PRI=nnnnnnn

  This is an optional parameter specifying the number of primary extent blocks to be used during allocation of temporary dataset OTDBS100 used during execution of the utility.   ‘nnnnnnn’ is a 1-7 digit numeric value in the range 1-8388607. If omitted, a default value of 10000 will be used. The length of each block will be equal to 100 times the concatenated field lengths of the table columns which are used to uniquely index each row in the table (see “SQLIN entry specification” on page 85 for a discussion of the unique index requirement).

- S100SEC=nnnnnnn

  This is an optional parameter specifying the number of secondary extent blocks to be used during allocation of temporary dataset OTDBS100 used during execution of the utility. ‘nnnnnnn’ is a 1-7 digit numeric value in the range 1-8388607. If omitted, a default value of 10000 will be used. The length of each block will be equal to 100 times the concatenated field lengths of the table columns which are used to uniquely index each row in the table (see “SQLIN entry specification” on page 85 for a discussion of the unique index requirement).

- S100UNIT=xxxxxxxx

  This is an optional parameter specifying the unit name to be used for allocation of temporary dataset OTDBS100 used during execution of the utility. ‘xxxxxxxx’ is any valid disk or VIO unit name. If omitted, a default value of ‘VIO’ is used.

- ROWMAX=nnnnnn|<u>0</u>

  This is an optional parameter specifying the maximum number of rows to be migrated from DB2 by the utility.   The utility will terminate processing at the first commit point (as controlled via the COMMIT parameter) after reaching this value. ‘nnnnnn’ must be a 1-6 digit numeric value in the range 0-999999. A value of 0 indicates that there is no maximum to the number of rows to be migrated during execution of the utility. If this parameter is omitted, a default value of 0 is used.

- TIME=nnnn|0

  This is an optional parameter specifying the maximum time for which execution of the utility is to continue. The utility will terminate processing at the first commit point (as controlled via the COMMIT parameter) after reaching this value. ‘nnnn’ must be a 1-4 digit numeric value in the range 0-999999. A value of 0 indicates that there is no maximum to the execution time of the utility. If this parameter is omitted, a default value of 0 is used.

SQLIN entry specification

The SQLIN file is a fixed-block sequential dataset with a logical record length of 80 bytes. It must contain one or more records containing a free-format text description of the rule(s) to be used for selection of objects to be migrated from DB2 to Archive Manager. Character positions 1-71 will be used for this description. Contents of character positions 72-80 in each record will be ignored. A maximum of 15 records may be present in the SQLIN file.

Rules for migration are specified in the format of a standard DB2-compatible SQL SELECT command, which must have the general format:

```
SELECT key1, key2, .... keyn FROM table_name
WHERE migration_criteria
```

where ‘key1’, ‘key2’, …. ‘keyn’ are the names of the columns which are used to establish a unique key for each row in the database, ‘table_name’ is the name of the DB2 table which is to be processed by the utility (ie. the name of the table itself, not the view of the table created during product installation processing), and ‘migration_criteria’ are the criteria specified by the user for selection of rows which are to be eligible for migration to Archive Manager (eg):

```
SELECT ACCOUNT, SUBCODE FROM OTDBTEST.TAB1
WHERE DATE < ’01.02.1997’
```

Note that migration criteria are specified using the standard DB2-compatible SQL criteria specification format.

A unique key must exist for each row in the database in order for it to be eligible for archival by the utility. The maximum number of key fields which may be specified is 8. Any valid SQL selection criteria may be entered in order to identify rows which are to be eligible for migration.

The archival utility will cause this statement to be validated by DB2 for correct SQL syntax. Any invalid specification will cause an SQL error to be received by the utility, which will cause it to terminate with a condition code of 12. In these circumstances, the SQL statement should be corrected and the job restarted.

PARMLIB requirements

The following ENVCNTL parameter library entries are used by the DB2 Manager control region during table migration processing. It should be verified that these are set as required prior to the start of table migration processing. If not, they may be altered via the DB2 Manager ‘SET’ operator command (see page 58):

SMFRECID This parameter should be set to the identifier of the SMF records to be created by DB2 Manager during table migration processing, or set to 0 if no SMF recording is to take place.

OBJSIZE This parameter should be set (or defaulted) to control the maximum number of migrated rows to be stored per Archive Manager object during table migration processing. Note that the lower of the two values for the ENVCNTL OBJSIZE parameter and the SYSIN COMMIT parameter is used to determine the maximum number of migrated rows per Archive Manager object.

WRITETIMEOUT This parameter will control the length of time that a writer task will wait between receiving successive requests when updating an Archive Manager database. If no request is received when this time interval expires, the writer task will disconnect the Archive Manager database without committing updates. This will cause the current tape volume to be dismounted, and the database to be left in an “uncommitted” condition. A rerun of the table migration job will then be required with the execution parameter “RESTART=YES” specified.

Print reports

The table migration utility will write a number of print reports to the SYSPRINT dataset during execution. These consist of:

- **Parameter specification and validation report.** This will contain a list of SYSIN parameters specified (or defaulted) for execution of the utility, and a validation error report if errors are detected in parameter specification. It will also specify the SQL migration selection statement coded in the SQLIN file.

- **Execution report.** Any errors or warning conditions encountered during execution of the utility will be identified in this report.

- **Summary processing report.** At end of processing, the utility will write the summary processing report which will identify the total number of table rows processed, and the total number migrated, up to the last successful commit point (for runs which do not terminate successfully) or up to end of processing (for successful runs).

Condition codes

The table migration utility will set a condition code on completion. This code may take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Utility failure and restart considerations

If the table migration utility fails to complete successfully (return code greater than 8), consult the error report in the SYSPRINT dataset to identify the cause of the problem. Refer to chapter 7 for a description of the message(s) appearing in the error report. All rows successfully processed prior to the last successful commit point before failure will have been successfully migrated from DB2 to Archive Manager, and any access requests for those objects will be satisfied by DB2 Manager from the appropriate Archive Manager database.

After correcting the error, the utility may be rerun with an execution parameter of “RESTART=YES” specified. The utility will automatically restart processing from the last successful commit point during the failed execution. No other special actions need to be taken prior to re-starting the failed table migration job.

Operator commands

Execution of the utility may be terminated at any point by entering the following command through the system console:

```
(STO)P jobname
```

where ‘jobname’ is the name of the job used to execute the object management procedure.

The utility will acknowledge receipt of this command via message OTD10023, and perform normal termination processing. All database updates performed by the utility prior to termination will be committed, and the utility will terminate with a condition code of 4.

Further executions of the table migration job for the appropriate DB2 table may subsequently be performed as normal.

■ Data management processing

Following successful execution of a table migration process, a number of follow-up activities may be required in order to complete the management of migrated and unmigrated data. These activities consist of the following:

- DB2 table backup and recovery
- DB2 table reorganization
- Archive Manager database backup and recovery

DB2 table backup and recovery

There is no change required to backup and recovery processing procedures for tablespaces containing tables which have been enabled for DB2 Manager migration. Image-copies should be taken normally as for non-enabled tables. However, as the size of the tablespaces used for storage of tables may now have been substantially reduced, the length of time taken for image-copy processing will be reduced by a corresponding amount.

Restore of a tablespace containing a migration-enabled table is performed in an identical manner to that of a tablespace containing a non-enabled table. DB2 will restore the 18-byte stubs containing information about migrated rows, as well as restoring non-migrated rows, to the condition they were in at the time the image-copy was taken. The stubs contain all the information necessary to access migrated rows from Archive Manager.

DB2 database reorganization

Tables which have been enabled for migration processing must be defined in a partitioned tablespace (see “Enabling a DB2 table for migration processing” on page 75). This tablespace will have one partition for exclusive storage of migrated row stubs, and one or more partitions for storage of non-migrated rows.

Database reorganization processing should **not** be performed on the partition containing the migrated row stubs, unless absolutely necessary. Reorganization of this partition will cause all migrated rows to be retrieved from Archive Manager. If Archive Manager objects are held on tape storage only, this is likely to generate a substantial amount of tape activity and could take an excessive length of time to complete. However, all rows stored in this partition will be a fixed 18 bytes in length. In this case, deletion of rows is not likely to cause fragmentation in the tablespace, as deleted space can be re-used as required. The requirement for reorganization of this partition is thus not mandatory.

Reorganization of other (non-migrated) partitions in the tablespace should be performed as normal. Regular reorganization will, in particular, be required for a period of time after implementation of DB2 Manager migration, in order to reclaim disk space occupied by rows which have been migrated to Archive Manager. This will allow these tablespace partitions to be reduced in size, eventually down to the size needed for the “working-set” of unmigrated rows (ie) rows which have yet to be migrated to Archive Manager.

Reorganization of one or more partitions of a partitioned tablespace is accomplished through the use of the PART keyword on the REORG TABLESPACE utility command. Refer to the appropriate DB2 Utility Guide and Reference manual for a complete description of this process.

Archive Manager database backup and recovery

Archive Manager database backup

After completion of the table migration process, standard Archive Manager database backup procedures should be implemented, as documented in the Archive Manager User Manual. The following Archive Manager database system files should be backed up, using standard disk backup procedures:

- Primary index dataset
- Storage level 0 space management dataset
- Journal dataset (optional – only required if journaling is enabled for the Archive Manager database).

The tape volume(s) updated or created during table migration processing should be backed up using the Archive Manager database backup utility OTASP110. Standard Archive Manager archival and backup processing enables the creation of up to 4 copies of each tape volume in an Archive Manager database. The copies are identified as follows:

‘A’ copy - this is the primary copy created when writing objects to an Archive Manager tape database. The primary copy will always exist.

‘B’ copy - this is a backup copy of the primary (‘A’) volume, created via execution of the Archive Manager backup utility (OTASP110).

‘C’ copy - this is a duplex copy of the primary (‘A’) volume. The ‘C’ copy is automatically created when writing objects to an Archive Manager database for which the duplexing facility has been enabled via the Archive Manager database administration facility. The duplex copy is created at the same time as the primary copy (ie) two tape drives will be allocated concurrently by Archive Manager.

‘D’ copy - this is a duplex copy of the backup (‘B’) volume. The ‘D’ copy is automatically created by the Archive Manager backup utility, when backing up a database for which the ‘duplex during backup’ facility has been enabled via the Archive Manager database administration facility. The backup duplex copy is created at the same time as the backup copy.

Each copy will contain a single standard-labeled dataset. The contents of each copy will be identical. The following dataset naming standards are used by Archive Manager to identify each copy in an Archive Manager tape database:

‘A’ copy:	‘hlq.Gnnnnn’
‘B’ copy:	‘hlq.Bs.Gnnnnn’
‘C’ copy:	‘hlq.Cs.Gnnnnn’
‘D’ copy:	‘hlq.Ds.Gnnnnn’

where ‘hlq’ is the high-level qualifier for system datasets, as defined during Archive Manager database definition processing (see “Define Archive Manager databases” on page 27).

‘s’ is the storage level (0-7)

and ‘nnnnn’ is the sequence number of the dataset within the Archive Manager database.

Objects are always written to storage level 0 when first migrated to tape. They may subsequently be moved to other storage levels via execution of the Archive Manager object management utility.

Archive Manager also provides a facility (via its database administration process) to automatically control the tape cartridge devices to be used to create each copy during the object migration or backup process

(eg) ‘A’ copy to primary library location. This copy of the database will be used for normal production access.

‘B’ copy to offsite location. This copy of the database will be used for disaster recovery purposes.

‘C’ copy to alternate onsite location. This copy of the database will be used for recovery after loss of an individual volume. This location may be in another library storage module, for automated recovery access, or on a free-standing device for eventual shelf storage.

Note that if duplexing during archival is enabled for the Archive Manager database, then a duplex tape volume will be created concurrently with the primary tape volume during database update processing. In some circumstances (e.g. if the duplex volume is created at a remote offsite location) it may not be necessary to execute the Archive Manager database backup utility. Refer to the Archive Manager User Manual for full information on these considerations.

Archive Manager database recovery

Archive Manager database recovery may be required in the following circumstances:

1. loss of an individual tape volume from with an Archive Manager tape database (eg) due to media failure

2. loss of an entire Archive Manager database (eg) due to loss of data center.

In both the above cases, standard Archive Manager database recovery procedures should be followed. A full description of these procedures may be found in the 'Recovery Control Processing' section in the Archive Manager User Manual.

The procedure in each case is summarized below:

1. <u>Loss of individual tape volume(s)</u>

   Identify the dataset sequence number(s) of the tape volume(s) which have become unavailable. A tape volume's dataset sequence number may be obtained from the name of the dataset contained on that volume. Refer to "Archive Manager database backup and recovery" on page 89 for a description of Archive Manager dataset naming standards.

   Set each sequence number (or range of sequence numbers) in recovery status, using the standard Archive Manager database administration dialog for the database containing the unavailable volumes. The copy to be used for recovery ('B', 'C' or 'D') is established during this process. Archive Manager will automatically access the appropriate backup copy when retrieving objects from all volumes which have been placed in recovery status.

   The primary copy of all volumes which are in recovery status should be re-created at a subsequent time using the Archive Manager database recovery utility OTASP130. This utility will automatically remove a volume from recovery status after its primary copy has been re-created. Volumes may be manually removed from recovery status at any time via the Archive Manager database administration dialog.

2. <u>Loss of entire database</u>

   If an entire database has been lost (due to hardware malfunction or loss of data center), the database must be placed in 'disaster recovery' status via the standard Archive Manager database administration dialog. The copy to be used for disaster recovery ('B', 'C' or 'D') is specified during this procedure.

   When a database is in disaster recovery status, all retrievals by Archive Manager from that database will be performed from the appropriate backup copy. In order to guarantee the continuance of established service levels for DB2 applications, all volumes in that copy of the database should be available for automated processing.

If disaster recovery is being performed on another processor, all disk datasets used by DB2 Manager and Archive Manager, and all user catalog entries from the primary facility, must be restored for use on the disaster recovery facility. There is no requirement for physical recovery of any tape dataset prior to disaster recovery usage.

If necessary, the primary copy of the database may be re-created using the standard Archive Manager database recovery utility (OTASP130). After a primary volume has been re-created, it will be removed from recovery status, and will be used thereafter to satisfy accesses to objects on that volume, as normal.

DB2 Manager Utilities

6

A number of batch utility programs are supplied with DB2 Manager, for provision of a range of facilities for supporting the DB2 Manager operational environment, and for extending functional capabilities supplied with other components of the product.

These utilities consist of:

- **OTDBP120** – the DB2 Manager row restore utility. This utility allows customers to identify migrated rows which are to be restored from Archive Manager to DB2 storage. The utility will retrieve selected rows from Archive Manager storage, and restore them in the DB2 table from which they were originally migrated.
- **OTDBP130** – the DB2 Manager pre-fetch utility. This utility allows customers to pre-fetch Archive Manager objects containing identified migrated rows from tape to disk storage within an Archive Manager database. The migration status of rows processed by this utility will remain unaltered.
- **OTDBP140** – the DB2 Manager table analysis utility. This scans a DB2 table and issues a print report giving a detailed and/or summary analysis of the contents of the table, identifying rows which have been migrated to Archive Manager storage.
- **OTDBP170** – the DB2 Manager database housekeeping utility. This scans a DB2 table and deletes any objects from the corresponding Archive Manager database which no longer hold active migrated rows. This will allow utilities supplied with the base Archive Manager product to reclaim space occupied by deleted objects.

■ OTDBP120 - the row restore utility

OTDBP120 is the DB2 Manager row restore utility program. The program is executed as a batch job in order to retrieve one or more migrated rows from an Archive Manager database and to return them to the DB2 table from which they were originally migrated. After restore, the rows will be implicitly deleted from the Archive Manager database, and will become resident in DB2 only. They will then be eligible for migration back to Archive Manager by the table migration utility OTDBP100 at some subsequent time.

The rows to be restored from Archive Manager are identified by the customer via search criteria in an SQL command, in an identical manner to the way in which rows are identified for selection for migration by OTDBP100. Execution of the program is controlled via a series of execution parameters.

A print report is issued by the utility giving details of the execution parameters, the row restore selection criteria and summary information about the rows restored from Archive Manager.

Functions

The DB2 Manager row restore utility performs the following functions:

- It will process and validate a series of execution parameters from a SYSIN file, and an SQLIN object selection parameter which will govern execution of the utility. The utility will then obtain information from DB2 about the table to be updated.
- It will scan the DB2 table, and select migrated rows for restore from the Archive Manager database which satisfy the selection criteria specified in the SYSIN dataset. An entry is written to an internal table for each selected row.
- After all rows in the table have been processed in this manner, the utility will sort all entries in the table into ascending database sequence. This will allow the utility to access the migrated rows in the most efficient manner, with the minimum amount of tape handling processing.
- The utility will process the sorted table sequentially. For each entry, the utility will update the corresponding table row by setting the archive indicator column (OTDBIND) back to ‘0’. This will cause the row to be retrieved from the Archive Manager database, and the 18-byte archive stub (in the table’s archive partition) to be replaced by the actual row (in one of the table’s non-archive partitions).
- Following successful update of a row, all subsequent access to that row will be satisfied from the DB2 table as normal. Access to the migrated row will have been lost. Storage space within the Archive Manager database for that row may be reclaimed by executing the DB2 Manager database housekeeping utility (as described on page 109).
- At the end of processing, a report will be produced giving the total number of rows which have been restored.

JCL requirements

The following JCL may be used to execute the utility:

```
//*
//STEP120  EXEC PGM=OTDBP120,REGION=4096K
//*
//STEPLIB   DD  DSN=Database.Manager.load.library,DISP=SHR
//          DD  DSN=Archive.Manager.load.library,DISP=SHR
//*
//OTASxxxx  DD  DSN=Archive.Manager.index.dataset,DISP=SHR
//SYSIN     DD  DSN=Execution.parameter.file,DISP=SHR
//SQLIN     DD  DSN=Row-restore.criteria.file,DISP=SHR
//SYSPRINT  DD  SYSOUT=*
//*
```

DD entries should be specified as follows:

STEPLIB: Enter the names of the DB2 Manager and Archive Manager product load libraries.

OTASxxxx: Enter the name of the primary index dataset of the Archive Manager database used for storage of objects from the DB2 table being processed by the utility. ‘xxxx’ should be set to the 4-character value used to identify the Archive Manager database in the DB2 Manager control region procedure (see “Perform MVS host system modifications” on page 21).

SYSPRINT: This entry is used for all print reports created during utility execution.

SYSIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 byte records, and is used for specification of the parameters used to control utility execution. Refer to “SYSIN parameter specification” on page 95 for a description of how to specify parameter entries in this file.

SQLIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 bytes records, and is used for specification of the criteria to be used by the utility for restoration of rows from Archive Manager to DB2. Refer to “SQLIN entry specification” on page 97 for a description of how to specify a restore criteria entry in this file.

SYSIN parameter specification

Parameters used to control execution of the utility are specified in file SYSIN. This must be a sequential dataset or in-stream JCL file containing one or more fixed-length 80-byte records, each record containing one or more separate parameter entries. Each parameter entry must be completely contained within one record (ie) continuation of any single parameter entry from one record to another is not permitted. A single record may contain multiple parameter entries, each being separated by one or more spaces, and optionally a single comma (‘,’).

A parameter entry has the general format:

```
keyword = value
```

Each keyword and its associated value are separated by an '=' sign, and optionally one or more spaces. The first keyword parameter in a record may begin in any character position from 1 onwards. Parameters may extend up to character position 71 of the record. Individual keyword parameters may not be continued over more than one record.

An asterisk ('*') in character position 1 will indicate a comment record, which will be ignored for parameter validation processing.

The following parameters may be specified:

- DB2SSID=xxxx

This mandatory parameter is used to specify the subsystem identifier of the DB2 subsystem to be used during execution of the utility. The subsystem identifier ‘xxxx’ can be 1-4 characters in length.

- PLAN=xxxxxxxx

  This is an optional parameter which is used to specify the DB2 plan name to be used during execution of the utility, where ‘xxxxxxxx’ is the 1-8 character identifier of the plan. If omitted, a default value of OTDBL120 is used. Refer to “Perform DB2 system modifications” on page 25 for information about DB2 Manager plan creation processing.

- COMMIT = nnnnn|0

  This is an optional parameter used to specify the frequency at which DB2 commit processing is to be performed by the utility. ‘nnnnn’ is a 1-5 digit numeric value in the range 0-99999, specifying the number of rows to be restored between each commit request. A value of 0 indicates that no explicit commit processing is to be performed by the program. If this parameter is omitted, a default value of 0 is used.

  Once commit processing has been successfully completed, all rows restored since the last commit point (or since the beginning of the job, for the first commit call) will have been successfully stored in the DB2 table, and all subsequent access to those rows will be performed via DB2 only.

- INDEXONLY=Y|N

  This optional parameter is used to control whether the selection criteria specified by the user for restoring rows are to be capable of resolution via table indexes only. The default value of “Y” indicates that the selection criteria must be capable of resolution via indexes only. If this is not the case, the program will issue an error message and terminate with a condition code 12. A value of “N” indicates that the selection criteria do not need to be capable of resolution via indexes only. In this case, execution of the row restore job may cause the utility to sequentially process all migrated rows from the Archive Manager database.

  If this parameter is omitted, the default value of “Y” is used.

- S120PRI=nnnnnnn

  This is an optional parameter specifying the number of primary extent blocks to be used during allocation of temporary dataset OTDBS120 used during execution of the utility. ‘nnnnnnn’ is a 1-7 digit numeric value in the range 1-8388607. If omitted, a default value of 10000 will be used. The length of each block will be equal to 100 times the concatenated field lengths of the table columns which are used to uniquely index each row in the table (see “SQLIN entry specification” on page 97 for a discussion of the unique index requirement).

- S120SEC=nnnnnnn

This is an optional parameter specifying the number of secondary extent blocks to be used during allocation of temporary dataset OTDBS120 used during execution of the utility. ‘nnnnnnn’ is a 1-7 digit numeric value in the range 1-8388607. If omitted, a default value of 10000 will be used. The length of each block will be equal to 100 times the concatenated field lengths of the table columns which are used to uniquely index each row in the table (see “SQLIN entry specification” on page 97 for a discussion of the unique index requirement).

- S120UNIT=xxxxxxxx

  This is an optional parameter specifying the unit name to be used for allocation of temporary dataset OTDBS120 used during execution of the utility. ‘xxxxxxxx’ is any valid disk or VIO unit name. If omitted, a default value of ‘VIO’ is used.

SQLIN entry specification

The SQLIN file is a fixed-block sequential dataset with a logical record length of 80 bytes. It must contain one or more records containing a free-format text description of the rule(s) to be used for selection of rows to be restored from Archive Manager to DB2. Character positions 1-71 will be used for this description. Contents of character positions 72-80 in each record will be ignored. A maximum of 15 records may be present in the SQLIN file.

Rules for row restore are specified in the format of a standard DB2-compatible SQL SELECT command, which must have the general format:

```
SELECT key1, key2, .... keyn FROM table_name
WHERE row_restore_criteria
```

where ‘key1’, ‘key2’, …. ‘keyn’ are the names of the columns which are used to establish a unique key for each row in the database, ‘table_name’ is the name of the DB2 table which is to be processed by the utility, (ie. the name of the table itself, not the view of the table created during product installation processing), and ‘row_restore_criteria’ are the criteria specified by the user for selection of rows which are to be restored to Archive Manager (eg):

```
SELECT ACCOUNT, SUBCODE FROM OTDBTEST.TAB1
WHERE ACNO = ‘123456789’
```

Note that row restore criteria are specified using the standard DB2-compatible SQL criteria specification format. Note also that where more than one unique index exists for the table, keys from any of these indexes may be used in the SELECT statement. In these circumstances, the choice of unique key(s) should be influenced by the search argument, in order to allow selection of rows for restore to be performed via INDEXONLY processing where possible.

The row restore utility will cause this statement to be validated by DB2 for correct SQL syntax. Any invalid specification will cause an SQL error to be received by the utility, which will cause it to terminate with a condition code of 12. In these circumstances, the SQL statement should be corrected and the job restarted.

Print reports

The row restore utility will write a number of print reports to the SYSPRINT dataset during execution. These consist of:

- **Parameter specification and validation report.** This will contain a list of SYSIN parameters specified (or defaulted) for execution of the utility, and a validation error report if errors are detected in parameter specification. It will also specify the SQL row restore selection statement coded in the SQLIN file.
- **Execution report.** Any errors or warning conditions encountered during execution of the utility will be identified in this report.
- **Summary processing report.** At end of processing, the utility will write the summary processing report which will identify the total number of table rows processed, and the total number restored, up to the last successful commit point (for runs which do not terminate successfully) or up to end of processing (for successful runs).

Condition codes

The row restore utility will set a condition code on completion. This code may take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Utility failure and restart considerations

If the row restore utility fails to complete successfully (return code greater than 8), consult the error report in the SYSPRINT dataset to identify the cause of the problem. Refer to chapter 7 for a description of the message(s) appearing in the error report. All rows successfully processed prior to the last successful commit point before failure will have been successfully restored from Archive Manager to DB2, and any access requests for those objects will be satisfied via DB2 only.

After correcting the error, the utility may be rerun normally. The utility will automatically restart processing from the last successful commit point during the failed execution. No special actions need to be taken prior to re-starting the failed row restore job.

Clean-up processing

Following successful execution of the row restore utility, the storage used in the Archive Manager database for rows which have been restored to DB2 will no longer be accessible, and will be eligible for re-use. However, this space cannot be re-used until the Archive Manager object holding a restored row has been deleted from the Archive Manager database.

As multiple migrated rows may be held in each Archive Manager object, it is not possible to delete the object until all migrated rows held therein have been invalidated (either due to being restored to DB2, or to the row being deleted from the DB2 table).

DB2 Manager provides a database housekeeping utility to perform cross-reference processing between a DB2 table and its associated Archive Manager database, and to delete Archive Manager objects which no longer contain active migrated rows. Standard Archive Manager object management and database maintenance processing can then be performed to allow re-use of index storage and tape and/or disk copy storage used for objects which have been restored. Refer to “Archive Manager database maintenance” on page 114 for a description of Archive Manager database housekeeping processing requirements.

■ OTDBP130 - the pre-fetch utility

OTDBP130 is the DB2 Manager pre-fetch utility program. The program is executed as a batch job in order to pre-fetch Archive Manager objects containing migrated rows from tape to disk storage within the Archive Manager database. After pre-fetching, the objects containing the selected rows will be accessed from disk instead of from tape, improving performance for access to migrated rows. Note that pre-fetching rows using this utility will not affect the migration status of the rows - these will remain in Archive Manager storage, and are not restored to the originating DB2 table. The row restore utility OTDBP120 should be used if it is required to return migrated rows to DB2 storage.

The rows to be pre-fetched by Archive Manager are identified by the customer via search criteria in an SQL command, in an identical manner to the way in which rows are identified for selection for migration by OTDBP100. Execution of the program is controlled via a series of execution parameters.

A print report is issued by the utility giving details of the execution parameters, the pre-fetch selection criteria and detailed and/or summary information about the rows pre-fetched by Archive Manager.

Disk copies of pre-fetched objects are retained by Archive Manager for the period of time specified in the database definition. Pre-fetched objects will be deleted by the Archive Manager object management utility following expiration of this time interval. Refer to the Lifecycle Director Archive Manager User Guide for information about setting pre-fetch retention periods in a database, and about executing the object management utility.

Functions

The DB2 Manager pre-fetch utility performs the following functions:

- It will process and validate a series of execution parameters from a SYSIN file, and an SQLIN row selection parameter which will govern execution of the utility. The utility will then obtain information from DB2 about the table from which the rows to be pre-fetched were migrated.
- It will scan the DB2 table, and select migrated rows for pre-fetching in the Archive Manager database which satisfy the selection criteria specified in the SYSIN dataset.
- For each selected row, a pre-fetch request will be passed to Archive Manager requesting that the object containing the selected row be copied from tape to disk, using the standard pre-fetch application program interface supplied by that component. If detailed print reporting has been requested an entry will be written to the report file giving details of the request passed to Archive Manager and the result of that request.
- When the number of requests accepted by Archive Manager has reached the batch size specified in the execution parameters, the program will issue a request to Archive Manager to process the batch. This will cause Archive Manager to sort all requests in the batch into the most efficient sequence for retrieval from tape, and to create disk copies of each object in the batch. Note that the same object may appear multiple times in each batch, if multiple rows within that object have been selected for pre-fetch processing. Archive Manager will ignore duplicate object requests in the batch.
- After all rows in the table have been processed in this manner, a report will be produced giving the total number of rows which have been pre-fetched.

JCL requirements

The following JCL may be used to execute the utility:

```
//*
//STEP130  EXEC PGM=OTDBP130,REGION=4096K
//*
//STEPLIB   DD  DSN=Database.Manager.load.library,DISP=SHR
//          DD  DSN=Archive.Manager.load.library,DISP=SHR
//*
//OTASxxxx  DD  DSN=Archive.Manager.index.dataset,DISP=SHR
//SYSIN     DD  DSN=Execution.parameter.file,DISP=SHR
//SQLIN     DD  DSN=Pre-fetch.criteria.file,DISP=SHR
//SYSPRINT  DD  SYSOUT=*
//*
```

DD entries should be specified as follows:

STEPLIB: Enter the names of the DB2 Manager and Archive Manager product load libraries.

OTASxxxx: Enter the name of the primary index dataset of the Archive Manager database used for storage of objects from the DB2 table being processed by the utility. ‘xxxx’ should be set to the 4-character value used to identify the Archive Manager database in the DB2 Manager control region procedure (see “Perform MVS host system modifications” on page 21).

SYSPRINT: This entry is used for all print reports created during utility execution.

SYSIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 byte records, and is used for specification of the parameters used to control utility execution. Refer to “SYSIN parameter specification” on page 101 for a description of how to specify parameter entries in this file.

SQLIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 bytes records, and is used for specification of the criteria to be used by the utility for pre-fetching of rows by Archive Manager. Refer to “SQLIN entry specification” on page 102 for a description of how to specify a pre-fetch criteria entry in this file.

SYSIN parameter specification

Parameters used to control execution of the utility are specified in file SYSIN. This must be a sequential dataset or in-stream JCL file containing one or more fixed-length 80-byte records, each record containing one or more separate parameter entries. Each parameter entry must be completely contained within one record (ie) continuation of any single parameter entry from one record to another is not permitted. A single record may contain multiple parameter entries, each being separated by one or more spaces, and optionally a single comma (‘,’).

A parameter entry has the general format:

```
keyword = value
```

Each keyword and its associated value are separated by an '=' sign, and optionally one or more spaces. The first keyword parameter in a record may begin in any character position from 1 onwards. Parameters may extend up to character position 71 of the record. Individual keyword parameters may not be continued over more than one record.

An asterisk ('*') in character position 1 will indicate a comment record, which will be ignored for parameter validation processing.

The following parameters may be specified:

- DB2SSID=xxxx

This mandatory parameter is used to specify the subsystem identifier of the DB2 subsystem to be used during execution of the utility. The subsystem identifier 'xxxx' can be 1-4 characters in length.

- PLAN=xxxxxxxx

  This is an optional parameter which is used to specify the DB2 plan name to be used during execution of the utility, where 'xxxxxxxx' is the 1-8 character identifier of the plan. If omitted, a default value of OTDBL130 is used. Refer to "Perform DB2 system modifications" on page 25 for information about DB2 Manager plan creation processing.

- BATCH = nnnnn|0

  This is an optional parameter used to specify the maximum number of pre-fetch requests which are to be processed by Archive Manager in a single batch. 'nnnnn' is a 1-5 digit numeric value in the range 0-99999, specifying the number of requests to be processed per batch. A value of 0 indicates that there is no maximum to the size of the request batch – all rows identified for pre-fetching during execution of the utility will be processed in a single batch. If this parameter is omitted, a default value of 0 is used.

- INDEXONLY=Y|N

  This optional parameter is used to control whether the selection criteria specified by the user for pre-fetching rows are to be capable of resolution via table indexes only. The default value of "Y" indicates that the selection criteria must be capable of resolution via indexes only. If this is not the case, the program will issue an error message and terminate with a condition code 12. A value of "N" indicates that the selection criteria do not need to be capable of resolution via indexes only. In this case, execution of the pre-fetch job may cause the utility to sequentially process all migrated rows from the Archive Manager database.

  If this parameter is omitted, the default value of "Y" is used.

- PRINT=DETAIL|SUMMARY

  This is an optional parameter which is used to control the level of reporting performed by the utility. If a value of SUMMARY is specified for this parameter, the detailed processing report will be omitted. All other print reports will be produced as normal. If the default value of DETAIL is specified, all print reports (including the detailed processing report) will be produced.

SQLIN entry specification

The SQLIN file is a fixed-block sequential dataset with a logical record length of 80 bytes. It must contain one or more records containing a free-format text description of the rule(s) to be used for selection of rows to be pre-fetched by Archive Manager. Character positions 1-71 will be used for this description.

Contents of character positions 72-80 in each record will be ignored. A maximum of 15 records may be present in the SQLIN file.

Pre-fetch rules are specified in the format of a standard DB2-compatible SQL SELECT command, which must have the general format:

```
SELECT key1, key2, .... keyn FROM table_name
WHERE pre-fetch_criteria
```

where ‘key1’, ‘key2’, …. ‘keyn’ are the names of the columns which are used to establish a unique key for each row in the database, ‘table_name’ is the name of the DB2 table which is to be processed by the utility, (ie. the name of the table itself, not the view of the table created during product installation processing), and ‘pre-fetch_criteria’ are the criteria specified by the user for selection of rows which are to be pre-fetched by Archive Manager (eg):

```
SELECT ACCOUNT, SUBCODE FROM OTDBTEST.TAB1
WHERE ACNO = ‘123456789’
```

Note that pre-fetch criteria are specified using the standard DB2-compatible SQL criteria specification format. Note also that where more than one unique index exists for the table, keys from any of these indexes may be used in the SELECT statement. In these circumstances, the choice of unique key(s) should be influenced by the search argument, in order to allow selection of rows for pre-fetching to be performed via INDEXONLY processing where possible.

The pre-fetch utility will cause this statement to be validated by DB2 for correct SQL syntax. Any invalid specification will cause an SQL error to be received by the utility, which will cause it to terminate with a condition code of 12. In these circumstances, the SQL statement should be corrected and the job restarted.

Print reports

The pre-fetch utility will write a number of print reports to the SYSPRINT dataset during execution. These consist of:

- **Parameter specification and validation report.** This will contain a list of SYSIN parameters specified (or defaulted) for execution of the utility, and a validation error report if errors are detected in parameter specification. It will also specify the SQL pre-fetch selection statement coded in the SQLIN file.
- **Execution report.** Any errors or warning conditions encountered during execution of the utility will be identified in this report.
- **Detail processing report.** If PRINT=DETAIL has been specified (or allowed to default) in the execution parameter file, a detail print line will be written for each pre-fetch request passed to Archive Manager. This will contain the following information:
  - Request type (add to batch, or process batch)
  - Archive Manager database identifier

- Primary key and archive date of the object containing the row to be pre-fetched
- Return and supplementary codes received from Archive Manager after processing the request
- The concatenated key of the DB2 row to be pre-fetched

Note that there may be multiple entries in this report for the same Archive Manager object, if multiple rows within that object have been selected for pre-fetch processing. Repeated requests for pre-fetching of the same object will be ignored by Archive Manager.

- **Summary processing report.** At end of processing, the utility will write the summary processing report. This will contain the following information:
  - Number of rows selected for pre-fetching
  - Number of DB2 error conditions received
  - Total number of pre-fetch requests issued
  - Number of pre-fetch requests accepted by Archive Manager
  - Number of rows which were already pre-fetched
  - Number of pre-fetch requests rejected by Archive Manager
  - Number of batches successfully processed
  - Number of batches processed with errors

Condition codes

The pre-fetch utility will set a condition code on completion. This code may take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Utility failure and restart considerations

If the pre-fetch utility fails to complete successfully (return code greater than 8), consult the error report in the SYSPRINT dataset to identify the cause of the problem. Refer to chapter 7 for a description of the message(s) appearing in the error report. Rows in all batches successfully processed prior to the point of failure will have been successfully pre-fetched to disk storage by Archive Manager.

After correcting the error, the utility may be rerun normally. No special actions need to be taken prior to re-starting the failed pre-fetch job.

Archive Manager housekeeping processing

Pre-fetched objects will be retained on disk in the Archive Manager database for the period of time specified in the definition of the database. When that time period has expired, the disk copies of pre-fetched objects will be eligible for deletion by Archive Manager, at which point any retrieval access for rows contained in those objects will be satisfied from tape.

Standard Archive Manager object management processing should be performed on a regular basis to delete the expired disk-copies of pre-fetched objects. Refer to “Archive Manager database maintenance” on page 114 for a description of Archive Manager database housekeeping processing requirements.

■ OTDBP140 - the table analysis utility

OTDBP140 is the DB2 Manager table analysis utility program. The program is executed as a batch job in order to produce a detailed and/or summary analysis of the rows in a DB2 table which has been enabled for row migration processing by DB2 Manager.

A print report is issued by the utility giving details of the execution parameters, summary information about the rows in the DB2 table, and optionally a detailed report on the contents of the table.

Functions

The DB2 Manager table analysis utility performs the following functions:

- It will process and validate a series of execution parameters from a SYSIN file, and an SQLIN table identification parameter which will govern execution of the utility. The utility will then obtain information from DB2 about the table to be analyzed.
- It will scan the DB2 table, and identify whether a row has been migrated to DB2 Manager or not. For each migrated row, it will optionally write an entry to the detailed processing report, identifying the row and its location in the associated Archive Manager database.
- At the end of processing, a summary processing report will be produced giving an analysis of table rows.

JCL requirements

The following JCL may be used to execute the utility:

```
//*
//STEP140  EXEC PGM=OTDBP140,REGION=4096K
```

```
//*
//STEPLIB   DD  DSN=Database.Manager.load.library,DISP=SHR
//          DD  DSN=Archive.Manager.load.library,DISP=SHR
//*
//OTASxxxx  DD  DSN=Archive.Manager.index.dataset,DISP=SHR
//SYSIN     DD  DSN=Execution.parameter.file,DISP=SHR
//SQLIN     DD  DSN=Table.identification.file,DISP=SHR
//SYSPRINT  DD  SYSOUT=*
//*
```

DD entries should be specified as follows:

STEPLIB: Enter the names of the DB2 Manager and Archive Manager product load libraries.

OTASxxxx: Enter the name of the primary index dataset of the Archive Manager database used for storage of objects from the DB2 table being analyzed by the utility. ‘xxxx’ should be set to the 4-character value used to identify the Archive Manager database in the DB2 Manager control region procedure (see “Perform MVS host system modifications” on page 21).

SYSPRINT: This entry is used for all print reports created during utility execution.

SYSIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 byte records, and is used for specification of the parameters used to control utility execution. Refer to “SYSIN parameter specification” on page 106 for a description of how to specify parameter entries in this file.

SQLIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 bytes records, and is used for specification of the table to be processed by the utility. Refer to “SQLIN entry specification” on page 107 for a description of how to specify an entry in this file.

SYSIN parameter specification

Parameters used to control execution of the utility are specified in file SYSIN. This must be a sequential dataset or in-stream JCL file containing one or more fixed-length 80-byte records, each record containing one or more separate parameter entries. Each parameter entry must be completely contained within one record (ie) continuation of any single parameter entry from one record to another is not permitted. A single record may contain multiple parameter entries, each being separated by one or more spaces, and optionally a single comma (‘,’).

A parameter entry has the general format:

```
keyword = value
```

Each keyword and its associated value are separated by an '=' sign, and optionally one or more spaces. The first keyword parameter in a record may begin in any character position from 1 onwards. Parameters may extend up to character position 71 of the record. Individual keyword parameters may not be continued over more than one record.

An asterisk ('*') in character position 1 will indicate a comment record, which will be ignored for parameter validation processing.

The following parameters may be specified:

- DB2SSID=xxxx
  This mandatory parameter is used to specify the subsystem identifier of the DB2 subsystem to be used during execution of the utility. The subsystem identifier ‘xxxx’ can be 1-4 characters in length.
- PLAN=xxxxxxxx

  This is an optional parameter which is used to specify the DB2 plan name to be used during execution of the utility, where ‘xxxxxxxx’ is the 1-8 character identifier of the plan. If omitted, a default value of OTDBL140 is used. Refer to “Perform DB2 system modifications” on page 25 for information about DB2 Manager plan creation processing.
- PRINT=<u>DETAIL</u>|SUMMARY

  This is an optional parameter which is used to control the level of reporting performed by the utility. If a value of SUMMARY is specified for this parameter, the detailed processing report will be omitted. All other print reports will be produced as normal. If the default value of DETAIL is specified, all print reports (including the detailed processing report) will be produced.

SQLIN entry specification

The SQLIN file is a fixed-block sequential dataset with a logical record length of 80 bytes. It will contain one or more records containing a free-format SQL SELECT command which will identify the DB2 table to be processed and the key fields in the table. Character positions 1-71 will be used for this description. Contents of character positions 72-80 in each record will be ignored. A maximum of 15 records may be present in this file.

The SQL SELECT command should have the general format:

```
SELECT key1, key2, .. .., keyn FROM table_name
```

where ‘key1’, ‘key2’, …., ‘keyn’ are the names of the columns which are used to establish a unique key for each row in the database, and ‘table_name’ is the name of the DB2 table which is to be processed by the utility (ie. the name of the table itself, not the view of the table created during product installation processing)

(eg):

```
SELECT ACCOUNT, SUBCODE FROM OTDBTEST.TAB1
```

The table analysis utility will cause this statement to be validated by DB2 for correct SQL syntax. Any invalid specification will cause an SQL error to be received by the utility, which will cause it to terminate with a condition code of 12. In these circumstances, the SQL statement should be corrected and the job restarted.

Print reports

The table analysis utility will write a number of print reports to the SYSPRINT dataset during execution. These consist of:

- **Parameter specification and validation report.** This will contain a list of execution parameters specified (or defaulted) in file SYSIN, and a validation error report if errors are detected in parameter specification. It will also specify the table identification statement coded in the SQLIN file.
- **Execution report.** Any errors or warning conditions encountered during execution of the utility will be identified in this report.
- **Summary processing report.** At end of processing, the utility will write the summary processing report which will identify the total number of rows in the table, the number of un-migrated rows and the number of migrated rows.
- **Detailed processing report.** If PRINT=DETAIL is specified (or defaulted) in the execution parameters, the utility will create an entry in this report for each migrated row in the table. This entry will contain the following information for each migrated row:
    - Concatenated DB2 key of row
    - Id of Archive Manager database containing migrated row
    - Database storage level containing migrated row
    - Key of Archive Manager object containing migrated row
    - Archive date of Archive Manager object containing migrated row
    - Start record number within Archive Manager object
    - Record count
    - Sequence number of tape containing start of object
    - Sequence number of tape containing end of object
    - Block id containing start of object
    - An indicator ('D') which is set if a disk copy of the object exists in the Archive Manager database

See Figure 5 below for an example of the detailed processing report.

```
(C) StorageTek Corp.                           Lifecycle Director          YYYY/MM/DD HH:MM:SS Page nnnn
                                                DB2 Manager V1.1
OTDBP140                                     Table Analysis Utility
                                                Processing report
Table: OTDBSYST.TAB1A
---------------------------------------------------------------------------------------------------------
DBid S Primary  Archive    Rec    Rec   Start End   Tape      D DB2 Concatenated
     L Key      Date       No     Count Seq   Seq   Block id    Key


ST01 0 000000A3 2003/03/06 00053 00001 00002 00002 02400150 D AC00001201002
ST01 0 000000A3 2003/03/06 00068 00001 00002 00002 02400150 D AC00002201002
ST01 0 000000A3 2003/03/06 00071 00001 00002 00002 02400150 D AC00003211002
ST01 0 000000A3 2003/03/06 00065 00001 00002 00002 02400150 D AC00006241002
...........
```

Figure 5: Detailed table analysis report

Condition codes

The table analysis utility will set a condition code on completion. This code may take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Utility failure and restart considerations

If the table analysis utility fails to complete successfully (return code greater than 8), consult the error report in the SYSPRINT dataset to identify the cause of the problem. Refer to chapter 7 for a description of the message(s) appearing in the error report.

After correcting the error, the utility may be rerun normally. The utility will start analysis processing from the beginning of the table once again. No special actions need to be taken prior to re-starting the failed table analysis job.

■ OTDBP170 - the database housekeeping utility

OTDBP170 is the DB2 Manager database housekeeping utility program. The program is executed as a batch job in order to scan a DB2 table and identify Archive Manager objects which no longer contain any active migrated rows. These objects will be deleted from the Archive Manager database used for storage of objects migrated from the DB2 table being processed by the utility. Migrated rows are deleted from an Archive Manager object in response to SQL UPDATE or DELETE commands issued by an application program, or by selection for processing by the DB2 Manager row restore utility.

The program will also “normalize” the storage of migrated rows in the DB2 table, if required, by ensuring that the stubs for migrated rows are only stored in the archive partition of the tablespace used for storage of the table, and that non-migrated rows are stored in one of the non-archive partitions of the tablespace. Any inconsistencies with the storage of migrated and non-migrated rows in the DB2 table will be resolved by the utility. This is required to ensure that re-organization of non-archive partitions can occur without reference to migrated rows in the Archive Manager database.

The table to be processed is identified by the customer via an SQL command. Execution of the program is controlled via a series of parameters in file SYSIN.

A print report is issued by the utility giving details of the execution parameters, and detailed and/or summary information about objects which have been deleted from Archive Manager. The detailed report will contain one entry for each object which has been deleted from the Archive Manager database. Each entry will contain the key of the Archive Manager object.

Functions

The DB2 Manager database housekeeping utility performs the following functions:

- It will process and validate a series of execution parameters from file SYSIN, and a table identification parameter from file SQLIN which will govern execution of the utility. The utility will then obtain information from DB2 about the table to be analyzed.
- It will sequentially scan the DB2 table identified in the SQLIN file, and for each migrated row will record information about the Archive Manager database object containing the migrated row. No access to the migrated rows themselves is required during the procedure.
- When all rows have been scanned, the utility will analyze the collected information to identify objects in the Archive Manager database which no longer contain any active migrated rows. It will issue a standard Archive Manager object deletion request for each such object. An entry may optionally be written to the detailed processing report for each deleted objects.
- At the end of processing, a report will be produced giving a summary analysis of the rows in the DB2 table and the objects deleted from the Archive Manager database.

JCL requirements

The following JCL may be used to execute the utility:

```
//*
//STEP170  EXEC PGM=OTDBP170,REGION=4096K
//*
//STEPLIB   DD  DSN=Database.Manager.load.library,DISP=SHR
//          DD  DSN=Archive.Manager.load.library,DISP=SHR
//*
//OTASxxxx  DD  DSN=Archive.Manager.index.dataset,DISP=SHR
//SYSIN     DD  DSN=Execution.parameter.file,DISP=SHR
//SQLIN     DD  DSN=Table.identification.file,DISP=SHR
//SYSPRINT  DD  SYSOUT=*
//*
```

DD entries should be specified as follows:

STEPLIB: Enter the names of the DB2 Manager and Archive Manager product load libraries.

OTASxxxx: Enter the name of the primary index dataset of the Archive Manager database used for storage of objects from the DB2 table being processed by the utility. ‘xxxx’ should be set to the 4-character value used to identify the Archive Manager database in the DB2 Manager control region procedure (see “Perform MVS host system modifications” on page 21).

SYSPRINT: This entry is used for all print reports created during utility execution.

SYSIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 byte records, and is used for specification of the parameters used to control utility execution. Refer to “SYSIN parameter specification” on page 112 for a description of how to specify parameter entries in this file.

SQLIN: This is a sequential dataset or in-stream JCL file containing fixed-length 80 bytes records, and is used for specification of the table to be processed by the utility. Refer to “SQLIN entry specification” on page 112 for a description of how to specify an entry in this file.

SYSIN parameter specification

Parameters used to control execution of the utility are specified in file SYSIN. This must be a sequential dataset or in-stream JCL file containing one or more fixed-length 80-byte records, each record containing one or more separate parameter entries. Each parameter entry must be completely contained within one record (ie) continuation of any single parameter entry from one record to another is not permitted. A single record may contain multiple parameter entries, each being separated by one or more spaces, and optionally a single comma (‘,’).

A parameter entry has the general format:

```
keyword = value
```

Each keyword and its associated value are separated by an '=' sign, and optionally one or more spaces. The first keyword parameter in a record may begin in any character position from 1 onwards. Parameters may extend up to character position 71 of the record. Individual keyword parameters may not be continued over more than one record.

An asterisk ('*') in character position 1 will indicate a comment record, which will be ignored for parameter validation processing.

The following parameters may be specified:

- DB2SSID=xxxx

This mandatory parameter is used to specify the subsystem identifier of the DB2 subsystem to be used during execution of the utility. The subsystem identifier ‘xxxx’ can be 1-4 characters in length.

- PLAN=xxxxxxxx

  This is an optional parameter which is used to specify the DB2 plan name to be used during execution of the utility, where ‘xxxxxxxx’ is the 1-8 character identifier of the plan. If omitted, a default value of OTDBL170 is used. Refer to “Perform DB2 system modifications” on page 25 for information about DB2 Manager plan creation processing.

- PRINT=DETAIL|SUMMARY

  This is an optional parameter which is used to control the level of reporting performed by the utility. If a value of SUMMARY is specified for this parameter, the detailed processing report will be omitted. All other print reports will be produced as normal. If the default value of DETAIL is specified, all print reports (including the detailed processing report) will be produced.

SQLIN entry specification

The SQLIN file is a fixed-block sequential dataset with a logical record length of 80 bytes. It will contain one or more records containing a free-format SQL SELECT command which will identify the DB2 table to be processed and the key fields in the table. Character positions 1-71 will be used for this description. Contents of character positions 72-80 in each record will be ignored. A maximum of 15 records may be present in this file.

The SQL SELECT command should have the general format:

```
SELECT key1, key2, .. .., keyn FROM table_name
```

where ‘key1’, ‘key2’, …., ‘keyn’ are the names of the columns which are used to establish a unique key for each row in the database, and ‘table_name’ is the name of the DB2 table which is to be processed by the utility (ie. the name of the table itself, not the view of the table created during product installation processing)

(eg):

```
SELECT ACCOUNT, SUBCODE FROM OTDBTEST.TAB1
```

The database housekeeping utility will cause this statement to be validated by DB2 for correct SQL syntax. Any invalid specification will cause an SQL error to be received by the utility, which will cause it to terminate with a condition code of 12. In these circumstances, the SQL statement should be corrected and the job restarted.

Print reports

The database housekeeping utility will write a number of print reports to the SYSPRINT dataset during execution. These consist of:

- **Parameter specification and validation report.** This will contain a list of execution parameters specified (or defaulted) in file SYSIN, and a validation error report if errors are detected in parameter specification. It will also specify the table identification statement coded in the SQLIN file.
- **Execution report.** Any errors or warning conditions encountered during execution of the utility will be identified in this report.
- **Summary processing report.** At end of processing, the utility will write the summary processing report which will identify the total number of rows in the table, the number of migrated rows, the number of rows normalized during processing (ie. wrongly located migrated rows which have been moved from a non-archive to the archive partition, and vice versa), and number of Archive Manager objects deleted.
- **Detailed processing report.** If PRINT=DETAIL is specified (or defaulted) in the execution parameters, the utility will create an entry in this report for each migrated row in the table. This entry will contain the following information for each migrated row:
  - Id of Archive Manager database containing deleted object
  - Archive Manager database storage level containing deleted object
  - Primary key of deleted object (4-byte hexadecimal value)
  - Archive date of deleted object
  - Archive Manager tape sequence number containing start of deleted object
  - Archive Manager tape sequence number containing end of deleted object
  - Identifier of tape block containing start of deleted object

  See Figure 6 for an example of the detailed processing report.

```
 (C) StorageTek Corp.                         Lifecycle Director          YYYY/MM/DD HH:MM:SS Page nnnn
                                               DB2 Manager V1.1
OTDBP170                                   Database Housekeeping Utility
                                               Processing report
TABLE: OTDBSYST.TAB1A
---------------------------------------------------------------------------------------------------------
DBid S Primary  Archive    Start End   Tape
     L Key      Date       Seq   Seq   Block


ST02 0 0000005C 2003/02/05 00001 00001 01400005
ST02 0 0000005D 2003/02/05 00001 00001 01400010
ST02 0 00000061 2003/02/12 00001 00001 014001A2
...........
```

Figure 6: Detailed processing report

Condition codes

The database housekeeping utility will set a condition code on completion. This code may take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Utility failure and restart considerations

If the database housekeeping utility fails to complete successfully (return code greater than 8), consult the error report in the SYSPRINT dataset to identify the cause of the problem. Refer to chapter 7 for a description of the message(s) appearing in the error report.

All deletes of Archive Manager objects completed up to the point of failure will be committed in the Archive Manager database. All DB2 updates performed for normalization of archive stub storage will be rolled back. DB2 updates are only committed on successful completion of the utility.

After correcting the error, the utility may be rerun normally. The utility will start housekeeping processing from the beginning of the table once again. No special actions need to be taken prior to re-starting the failed database housekeeping job.

Archive Manager database maintenance

Following successful completion of the DB2 Manager database housekeeping utility, the Archive Manager database used for storage of migrated rows from the processed DB2 table will contain one or more deleted objects. In order to recover the index and tape and disk storage space occupied by deleted objects, the Archive Manager database maintenance procedure must be performed. Maintenance of an Archive Manager database is performed by executing the following Archive Manager utilities:

- OTASP040 – the object management utility. This will recover the index entries for deleted objects and update tape residency statistics to allow tape recycling to take place.
- OTASP030 – the database maintenance utility. This will recycle data from all tape volumes in one or more storage levels of an Archive Manager database whose level of active objects falls below a user-defined threshold.

Refer to the Archive Manager User Manual for a full description of this utility and of its use in the Archive Manager database maintenance procedure. Note carefully that automatic expiration of Archive Manager objects via this utility should not be permitted, by letting the data retention period default to zero values when defining the Archive Manager database. Refer to “Define Archive Manager databases” on page 27 for information on this issue.

The database maintenance procedure should be performed at regular intervals for each Archive Manager database used for storage of migrated rows by DB2 Manager, in order to recover unused space from within the database which has been freed due to object deletion. The frequency at which the procedure needs to be performed will depend on the rate of object deletion, and the level of requirement for reclaiming tapes from the database in order to replenish scratch pools. The rate of object deletion, in turn, is governed by the frequency of deletion of migrated rows, either through explicit SQL DELETE request, SQL UPDATE requests, or restore of migrated rows from Archive Manager to DB2 via the row restore utility OTDBP120.

Control of the tape recycling process for an Archive Manager database is via the ‘recycle threshold’ entry set during the database definition process (see “Define Archive Manager databases” on page 27). This process must be executed for each DB2 table to be processed by DB2 Manager.

The recycle threshold specifies a numeric value in the range 00-99, which represents a percentage of active objects for any volume in the database. A tape volume becomes eligible for recycling by the Archive Manager database maintenance utility if the level of active objects on that volume falls below this threshold. If this entry is omitted during the database definition process, a default value of ‘0’ is used; this indicates that no recycling is to be performed on this database.

When a tape volume becomes eligible for recycling, the Archive Manager database maintenance utility will write all active objects contained within it to a new tape volume at the end of the database. The recycled volume will then be released from the database and may be returned to the appropriate scratch pool.

If the ‘autouncat during recycle’ entry has been set during the database definition process, released volumes will be automatically uncataloged after being released from the database. This facility may be used in conjunction with an installed tape management system to automatically return uncataloged volumes to the correct scratch pool.

The Archive Manager database maintenance utility processes a single Archive Manager database for each execution. Any tape volume being used by the database maintenance utility will be unavailable for allocation for row migration or migrated row access processing. It is recommended therefore that database maintenance for an Archive Manager database be scheduled for execution when there is no row recall or migration activity for the DB2 table associated with that Archive Manager database. It is not necessary to terminate the DB2 Manager control region while Archive Manager database maintenance is being performed.

An Archive Manager database’s recycle threshold may be altered at any time, via the supplied Archive Manager database administration dialogs. Note also that where multiple storage levels are in use for an Archive Manager database, a different recycle threshold may be supplied for each storage level.

As Archive Manager uses a discrete set of tape cartridge volumes and disk datasets for each database, it is possible to perform the Archive Manager database maintenance procedure on multiple DB2 Manager databases concurrently. Each procedure will require a minimum of two tape cartridge drives.

Refer to the Archive Manager User Manual for a complete description of the Archive Manager object management and database maintenance utilities and Archive Manager database housekeeping procedures.

This page is intentionally left blank

Messages and Codes

7

DB2 Manager components communicate with the user through informational, warning and error messages. All DB2 Manager messages begin with the characters 'OTD', followed by a 5-character numeric identifier. DB2 Manager messages may be displayed on the system log or in an DB2 Manager print report.

DB2 Manager SQL interface processing (for storage, retrieval and deletion of migrated rows) sets return and reason codes, using an SQL –652 error code, and a two-byte reason code set in field SQLERRD(6) of the SQL Communication Area (SQLCA).

This chapter documents all messages, categorized by DB2 Manager component, and all SQL reason codes issued by DB2 Manager.

■ OTD100xx – table migration utility messages

Messages from the DB2 Manager table migration utility will be written to the SYSPRINT dataset or the system log.

OTD10000 Row Migration Utility ended with completion code xx

Explanation: The DB2 Manager row migration utility has terminated. ‘xx’ gives the job completion code. This can take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Action: None. This is an informational message.

OTD10001 OPEN failed for xxxxxxxx dataset

Explanation: An attempt to open a non-VSAM dataset has failed. ‘xxxxxxxx’ identifies the filename assigned by the program to the dataset. Execution of the utility will be terminated with condition code 12.

Action: Consult other available system messages for further information on the cause of the open failure. Take the appropriate corrective action and re-run the utility.

OTD10002 The operand for keyword xxxxxxxx in exec parameters is too long

Explanation: The operand specified for the SYSIN parameter keyword ‘xxxxxxxx’ identified in the message is longer than the maximum value permitted. The character string ‘xxxxxxxx’ can take one of the following values: RESTART, DB2SSID, PLAN, COMMIT, S100PRI, S100SEC, NEWTAPE, LCDDBID. Execution of the utility will terminate with condition code 12.

Action: Refer to page 82 for a description of SYSIN parameter formats for the row migration utility. Correct the parameter error and re-submit the job.

OTD10003 Mandatory exec parameters have not been specified

Explanation: One or more mandatory parameters have not been specified in the SYSIN parameters used to control utility execution. Execution of the utility will be terminated with condition code 12.

Action: Refer to on page 82 for a description of SYSIN parameters for the row migration utility. Identify the missing mandatory parameter and re-submit the job.

OTD10004 Invalid or duplicate exec | SQLIN parameter. Please refer to the User Manual for valid specifications

Explanation: A SYSIN execution parameter or SQLIN parameter entry is invalid or has been specified more than once. Execution of the utility is terminated with condition code 12.

Action: Refer to page 80 for a description of valid SYSIN or SQLIN parameter entries. Correct the error and re-submit the job.

OTD10005 Load failed for program DSNALI | DSNHLI2.

Explanation: An attempt to dynamically load the identified program has failed. Execution of the utility is terminated with condition code 12.

Action: This message indicates that DB2 run-time modules were unavailable at time of execution. Ensure that the DB2 run-time load library is available to the executing job, and re-submit the job.

OTD10006 DB2 CAF error occurred during CONNECT |OPEN |CLOSE | DISCONNECT processing. Return Code = xxx, Reason Code = X'yyyyyyyy'

Explanation: A DB2 Call Attachment Facility error has occurred attempting to perform the type of processing identified in the message. 'xxx' gives the return code and 'yyyyyyyy' the hexadecimal reason code returned by the Call Attachment Facility. Execution of the utility is terminated with condition code 12.

Action: Refer to the DB2 Application Programming and SQL Guide for a description of CAF error and reason codes. Take the appropriate corrective action and re-start the utility.

OTD10007 The LCDDBID specified in the exec parameters is not consistent with the DB2 table name specified in the SQLIN statements

Explanation: The Archive Manager database identifier specified in the SYSIN execution parameters does not match that used previously for migration of rows from the DB2 table specified in the SQLIN file. Execution of the utility is terminated with condition code 12.

Action: Check that the correct values have been specified for the LCDDBID parameter in the SYSIN parameters, and the table name specified in the SQLIN file. Correct the invalid entry and re-submit the job.

If no error can be found in the parameter specifications, check the row entry in the LCDDB2.REGISTRY table for the table specified in the SYSIN parameters. The LCDDBID column in this row contains the Archive Manager database identifier associated with this table. The DB2 Manager row migration utility may only be executed if this combination of DB2 table and Archive Manager database identifier is specified in the parameters used to execute the utility. Ensure that the parameters supplied to the utility are consistent with the entry in the LCDDB2.REGISTRY table.

OTD10008 DB2 Table contains Large Objects (LOBs) or 'distinct types'

Explanation: The DB2 table specified in the SQLIN file cannot be processed, because it contains large objects (LOBs) or 'distinct types'. Tables containing these entities are not eligible for migration using DB2 Manager. Execution of the utility is terminated with a condition code 12.

Action: Check that the correct DB2 table name has been specified in the SQLIN file. If not, correct the entry and re-submit the job.

OTD10009 Key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx was not found in DB2 table

Explanation: The column name identified in the message has been specified as a key field in the SQLIN file, but the column does not exist in the DB2 table. Execution of the utility is terminated with condition code 12.

Action: Check that the names of the columns which uniquely index each row in the table have been specified correctly in the SQLIN file, and re-submit the job.

OTD10010 SQL Error during execution of xxxxxxxx command SQLCODE = xxxx

Explanation: An error has been received executing the SQL command identified in the message. Processing of the utility is terminated with a condition code 12. All DB2 and Archive Manager updates performed by the utility will be rolled-back to the last commit point. The following SQL commands may appear in this message: DESCRIBE, PREPARE, DECLARE, SET, EXECUTE, OPEN, FETCH, CLOSE, COMMIT. The message will be followed by a line containing parameters from the appropriate SQL error message.

Action: Refer to the DB2 Messages and Codes manual for a description of the SQL error code displayed in the message. Take the appropriate corrective action and re-submit the job, with SYSIN parameter “RESTART=YES” specified.

OTD10011 Return code X'xx’ retrieving Name/Token pair

Explanation: An error has occurred retrieving an OS/390 name/token pair. ‘xx’ gives the hexadecimal reason code received from the retrieve request. Execution of the utility is terminated with condition code 12.

Action: Refer to IBM documentation on the IEANTRT callable service for a description of the error code. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be identified, contact your local STK support representative.

OTD10012 Error locating ODCT

Explanation: The utility has been unable to locate the DB2 Manager control region communications table (ODCT) in ECSA storage. Execution of the utility is terminated with condition code 12.

Action: To resolve this error, terminate and re-start the DB2 Manager control region, then re-submit the job. If the error persists, contact your local STK support representative.

OTD10013 Return code ‘xx’ received during execution of TCBTOKEN | LOCASCB | ENQ macro.

Explanation: An error has been received attempting to execute the system macro identified in the message. ‘xx’ is the return code received from the macro call. Execution of the utility is terminated with condition code 12.

Action: Consult IBM system macro documentation for a description of the error. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD10014 Error X'xxxx’ Info X'yyyy’ dynamically allocating DDname OTDBS100

Explanation: An error has been received attempting to dynamically allocate temporary file OTDBS100. ‘xxxx’ and ‘yyyy’ give the hexadecimal error and information codes returned from the dynamic allocation request. Execution of the utility is terminated with condition code 12.

Action: Refer to IBM documentation on dynamic allocation (SVC99) processing for a description of the error and information codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD10015 The control region is not active

Explanation: An attempt has been made to run the row migration utility while the DB2 Manager is not started. Execution of the utility cannot continue.

Action: Ensure that the DB2 Manager control region is started before attempting to run the utility.

OTD10016 Normal run requested but restart run required

Explanation: RESTART=NO has been specified or defaulted in the SYSIN execution parameters specified for the jobstep, but the previous attempt to migrate rows to the Archive Manager database specified in the LCDDBID execution parameter failed. Execution of the utility is terminated with condition code 12.

Action: Check that the LCDDBID execution parameter has been specified correctly. If it has, RESTART=YES must be specified in the SYSIN parameters. Correct the SYSIN parameters as required, and re-submit the job.

OTD10017 Unable to obtain ODRE request element

Explanation: The utility has been unable to obtain an DB2 Manager control region request element from ECSA storage, in order to pass a request to the control region. Execution of the utility is terminated with condition code 12.

Action: This can occur if more than 1024 concurrent migrated row storage and retrieval requests are active in the system. If this is the case, re-schedule the row migration job to avoid reaching this limit. If this is not the case, try stopping and re-starting the DB2 Manager control region, and re-submitting the job. If the problem still persists, contact your local STK support representative.

OTD10018 Timeout error occurred when acquiring the scheduler lock

Explanation: The utility has been unable to obtain exclusive access to the control region scheduler lock within a time interval of 1 minute. Execution of the utility is terminated with condition code 12.

Action: This condition may occur if the DB2 Manager control region is exceptionally busy. If this is the case, re-schedule the row migration job at a less busy time. If this is not the case, examine the control region job log for exceptional messages which may give more information about the abnormal condition. Contact your local STK support representative if the problem cannot be resolved.

OTD10019 Timeout error occurred while waiting on response from the control region.

Explanation: The utility has passed a request to the control region, but a response was not received within the timeout limit specified in the READTIMEOUT parameter in the ENVCNTL parameter library member. Execution of the utility is terminated with condition code 12.

Action: Examine the job log for the DB2 Manager control region to determine if any abnormal condition has occurred which prevented the control region from responding to the request. If no obvious reason for the timeout can be identified, contact your local STK support representative.

OTD10020 Return code xx, supp codes X'yyyyyyyy' received during database OPEN | COMMIT | CLOSE processing.

Explanation: An error has been received from an Archive Manager open, commit or close request. 'xx' gives the return code and 'yyyyyyyy' the hexadecimal supplementary reason codes received from Archive Manager. Execution of the utility has been terminated with condition code 12.

Action: Refer to the Archive Manager User Manual for a description of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious reason for the error can be identified, contact your local STK support representative.

OTD10021 Insufficient storage for workareas

Explanation: An attempt by the utility to allocated virtual storage has failed Execution of the utility will be terminated with condition code 12.

Action: Increase the region size specified for execution of the utility and re-submit the job.

OTD10022 Table migration utility terminating in response to operator request

Explanation: This message is displayed on the system console to acknowledge receipt of an operator STOP command by the utility. Execution of the utility will terminate with condition code 4.

Action: None. This is an informational message.

OTD10023 Execution terminated in response to operator request

Explanation: This message is displayed at jobstep completion when execution of the utility has been terminated via an operator STOP command.

Action: None. This is an informational message.

OTD10024 Non-index fields specified in SELECT statement in SQLIN file

Explanation: One or more of the columns specified for retrieval in the SELECT statement in the SQLIN file do not appear in the unique table index. Execution of the utility is terminated with condition code 12.

Action: The columns specified in the SELECT statement must uniquely index each row in the table (see page 85 for a description of SQLIN file specification). Correct the SQLIN entry and re-submit the job.

OTD120xx – row restore utility messages

Messages from the DB2 Manager row restore utility will be written to the SYSPRINT dataset.

OTD12000 Row Restore Utility ended with completion code xx

Explanation: The DB2 Manager row restore utility has terminated. ‘xx’ gives the job completion code. This can take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Action: None. This is an informational message.

OTD12001 OPEN failed for xxxxxxxx dataset

Explanation: An attempt to open a non-VSAM dataset has failed. ‘xxxxxxxx’ identifies the filename assigned by the program to the dataset. Execution of the utility will be terminated with condition code 12.

Action: Consult other available system messages for further information on the cause of the open failure. Take the appropriate corrective action and re-run the utility.

OTD12002 The operand for keyword xxxxxxxx in exec parameters is too long

Explanation: The operand specified for the SYSIN parameter keyword ‘xxxxxxxx’ identified in the message is longer than the maximum value permitted. The character string ‘xxxxxxxx’ can take one of the following values: INDEXONLY, DB2SSID, PLAN, COMMIT, S120PRI, S120SEC, S120UNIT. Execution of the utility will terminate with condition code 12.

Action: Refer to page 95 for a description of SYSIN parameter formats for the row restore utility. Correct the parameter error and re-submit the job.

OTD12003 Key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx was not found in DB2 table

Explanation: The column name identified in the message has been specified as a key field in the SQLIN file, but the column does not exist in the DB2 table. Execution of the utility is terminated with condition code 12.

Action: Check that the names of the columns which uniquely index each row in the table have been specified correctly in the SQLIN file, and re-submit the job.

OTD12004 Mandatory exec parameters have not been specified

Explanation: One or more mandatory parameters have not been specified in the SYSIN execution parameters. Execution of the utility will be terminated with condition code 12.

Action: Refer to page 95 of this manual for a description of SYSIN parameters for the row restore utility. Identify the missing mandatory parameter and re-submit the job.

OTD12005 Invalid or duplicate exec | SQLIN parameter. Please refer to the User Manual for valid specifications

Explanation: A SYSIN parameter, or SQLIN parameter entry is invalid or has been specified more than once. Execution of the utility is terminated with condition code 12.

Action: Refer to page 93 manual for a description of valid SYSIN or SQLIN parameter entries. Correct the error and re-submit the job.

OTD12006 Insufficient storage for workareas

Explanation: An attempt by the utility to allocated virtual storage has failed Execution of the utility will be terminated with condition code 12.

Action: Increase the region size specified for execution of the utility and re-submit the job.

OTD12007 Load failed for program DSNALI | DSNHLI2

Explanation: An attempt to dynamically load the identified program has failed. Execution of the utility is terminated with condition code 12.

Action: This message indicates that DB2 run-time modules were unavailable at time of execution. Ensure that the DB2 run-time load library is available to the executing job, and re-submit the job.

OTD12008 Return code ‘xx’ received during execution of TCBTOKEN | ENQ macro

Explanation: An error has been received attempting to execute the system macro identified in the message. ‘xx’ is the return code received from the macro call. Execution of the utility is terminated with condition code 12.

Action: Consult IBM system macro documentation for a description of the error. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD12009 DB2 CAF error occurred during CONNECT | OPEN | CLOSE | DISCONNECT processing. Return Code = xxx, Reason Code = X'yyyyyyyy'

Explanation: A DB2 Call Attachment Facility error has occurred attempting to perform the type of processing identified in the message. 'xxx' gives the return code and 'yyyyyyyy' the hexadecimal reason code returned by the Call Attachment Facility. Execution of the utility is terminated with condition code 12.

Action: Refer to the DB2 Application Programming and SQL Guide for a description of CAF error and reason codes. Take the appropriate corrective action and re-start the utility.

OTD12010 SQL Error during execution of xxxxxxxx command SQLCODE = xxxx

Explanation: An error has been received executing the SQL command identified in the message. Processing of the utility is terminated with a condition code 12. All DB2 and Archive Manager updates performed by the utility will be rolled-back to the last commit point. The following SQL commands may appear in this message: DESCRIBE, PREPARE, DECLARE, SET, EXECUTE, OPEN, FETCH, CLOSE, COMMIT. The message will be followed by a line containing parameters from the appropriate SQL error message.

Action: Refer to the DB2 Messages and Codes manual for a description of the SQL error code displayed in the message. Take the appropriate corrective action and re-submit the job. No other special actions are required.

OTD12011 DB2 Table contains Large Objects (LOBs) or 'distinct types'

Explanation: The DB2 table specified in the SQLIN file cannot be processed, because it contains large objects (LOBs) or 'distinct types'. Tables containing these entities are not eligible for migration using DB2 Manager. Execution of the utility is terminated with a condition code 12.

Action: Check that the correct DB2 table name has been specified in the SQLIN file. If not, correct the entry and re-submit the job.

OTD12013 Retrieval using the specified KEY fields cannot be made via INDEXONLY processing

Explanation: Identification of objects to be restored from Archive Manager to DB2 cannot be performed via index-only processing, and a SQLIN parameter of INDEXONLY=Y has been specified (or defaulted). Execution of the utility cannot continue, and is terminated with a condition code 12.

Action: Refer to a description of the INDEXONLY execution parameter on page 95 for a discussion of considerations regarding index-only processing by the row restore utility. If necessary, change the setting of the INDEXONLY parameter and re-submit the job.

OTD12014 Error X'xxxx' Info X'yyyy' dynamically allocating file SORTIN | SORTOUT

Explanation: An error has been received attempting to dynamically allocate the temporary SORTIN or SORTOUT file used by the utility for internal sort requirements. ‘xxxx’ and ‘yyyy’ give the hexadecimal error and information codes returned from the dynamic allocation request. Execution of the utility is terminated with condition code 12.

Action: Refer to IBM documentation on dynamic allocation (SVC99) processing for a description of the error and information codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD12016 Return Code xx, Supp codes X'yyyyyyyy' received during OPEN | QUERY | CLOSE processing for Archive Manager database zzzz

Explanation: An error has been received from an Archive Manager open, query or close request. ‘xx’ gives the return code and ‘yyyyyyyy’ the hexadecimal supplementary reason codes received from Archive Manager. Execution of the utility has been terminated with condition code 12.

Action: Refer to the Archive Manager User Manual for a description of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious reason for the error can be identified, contact your local STK support representative.

OTD12017 Condition Code xx returned from internal sort - examine SYSOUT dataset for further information

Explanation: A non-zero condition code of ‘xx’ has been returned from an internal sort request issued by the utility. Processing of the utility is terminated with condition code 12.

Action: Refer to documentation for the sort product in use on the system on which the job was executing, for a description of the condition code displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious reason for the error can be identified, contact your local STK support representative.

■ OTD130xx – pre-fetch utility messages

Messages from the DB2 Manager pre-fetch utility will be written to the SYSPRINT dataset.

OTD13000 Batch Pre-fetch Utility ended with completion code xx

Explanation: The DB2 Manager pre-fetch utility has terminated. 'xx' gives the job completion code. This can take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Action: None. This is an informational message.

OTD13001 OPEN failed for xxxxxxxx dataset

Explanation: An attempt to open a non-VSAM dataset has failed. 'xxxxxxxx' identifies the filename assigned by the program to the dataset. Execution of the utility will be terminated with condition code 12.

Action: Consult other available system messages for further information on the cause of the open failure. Take the appropriate corrective action and re-run the utility.

OTD13002 The operand for keyword xxxxxxxx in exec parameters is too long

Explanation: The operand specified for the SYSIN parameter keyword 'xxxxxxxx' identified in the message is longer than the maximum value permitted. The character string 'xxxxxxxx' can take one of the following values: INDEXONLY, DB2SSID, PLAN, BATCH, PRINT. Execution of the utility will terminate with condition code 12.

Action: Refer to page 102 for a description of SYSIN parameter formats for the row restore utility. Correct the parameter error and re-submit the job.

OTD13003 Key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx was not found in DB2 table

Explanation: The column name identified in the message has been specified as a key field in the SQLIN file, but the column does not exist in the DB2 table. Execution of the utility is terminated with condition code 12.

Action: Check that the names of the columns which uniquely index each row in the table have been specified correctly in the SQLIN file, and re-submit the job.

OTD13004 Mandatory exec parameters have not been specified

Explanation: One or more mandatory parameters have not been specified in the SYSIN execution parameters. Execution of the utility will be terminated with condition code 12.

Action: Refer to page 101 of this manual for a description of SYSIN parameters for the row restore utility. Identify the missing mandatory parameter and re-submit the job.

OTD13005 Invalid or duplicate exec | SQLIN parameter. Please refer to the User Manual for valid specifications

Explanation: A SYSIN parameter, or SQLIN parameter entry is invalid or has been specified more than once. Execution of the utility is terminated with condition code 12.

Action: Refer to page 101 of the manual for a description of valid SYSIN or SQLIN parameter entries. Correct the error and re-submit the job.

OTD13006 Insufficient storage for workareas

Explanation: An attempt by the utility to allocated virtual storage has failed Execution of the utility will be terminated with condition code 12.

Action: Increase the region size specified for execution of the utility and re-submit the job.

OTD13007 Load failed for program DSNALI | DSNHLI2

Explanation: An attempt to dynamically load the identified program has failed. Execution of the utility is terminated with condition code 12.

Action: This message indicates that DB2 run-time modules were unavailable at time of execution. Ensure that the DB2 run-time load library is available to the executing job, and re-submit the job.

OTD13008 Return code ‘xx’ received during execution of TCBTOKEN | ENQ macro

Explanation: An error has been received attempting to execute the system macro identified in the message. ‘xx’ is the return code received from the macro call. Execution of the utility is terminated with condition code 12.

Action: Consult IBM system macro documentation for a description of the error. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD13009 DB2 CAF error occurred during CONNECT | OPEN | CLOSE | DISCONNECT processing. Return Code = xxx, Reason Code = X'yyyyyyyy’

Explanation: A DB2 Call Attachment Facility error has occurred attempting to perform the type of processing identified in the message. ‘xxx’ gives the return code and ‘yyyyyyyy’ the hexadecimal reason code returned by the Call Attachment Facility. Execution of the utility is terminated with condition code 12.

Action: Refer to the DB2 Application Programming and SQL Guide for a description of CAF error and reason codes. Take the appropriate corrective action and re-start the utility.

OTD13010 SQL Error during execution of xxxxxxxx command SQLCODE = xxxx

Explanation: An error has been received executing the SQL command identified in the message. Processing of the utility is terminated with a condition code 12. The following SQL commands may appear in this message: DESCRIBE, PREPARE, DECLARE, SET, EXECUTE, OPEN, FETCH, CLOSE, COMMIT. The message will be followed by a line containing parameters from the appropriate SQL error message.

Action: Refer to the DB2 Messages and Codes manual for a description of the SQL error code displayed in the message. Take the appropriate corrective action and re-submit the job. No other special actions are required.

OTD13011 DB2 Table contains Large Objects (LOBs) or 'distinct types’

Explanation: The DB2 table specified in the SQLIN file cannot be processed, because it contains large objects (LOBs) or ‘distinct types’. Tables containing these entities are not eligible for migration using DB2 Manager. Execution of the utility is terminated with a condition code 12.

Action: Check that the correct DB2 table name has been specified in the SQLIN file. If not, correct the entry and re-submit the job.

OTD13013 Retrieval using the specified KEY fields cannot be made via INDEXONLY processing

Explanation: Identification of objects to be pre-fetch by Archive Manager cannot be performed via index-only processing, and a SQLIN parameter of INDEXONLY=Y has been specified (or defaulted). Execution of the utility cannot continue, and is terminated with a condition code 12.

Action: Refer to a description of the INDEXONLY execution parameter on page 101 for a discussion of considerations regarding index-only processing by the row restore utility. If necessary, change the setting of the INDEXONLY parameter and re-submit the job.

OTD13015 Archive Manager is not at the correct level

Explanation: Archive Manager v2.6 or higher must be installed in order to be able to perform pre-fetch processing using the DB2 Manager pre-fetch utility. The version of Archive Manager used during execution of the utility is not at the correct level.

Action: Ensure that the correct Archive Manager load library has been specified in the STEPLIB, JOBLIB or linklist library concatenations, and rerun the job.

OTD13016 Return Code xx, Supp codes X'yyyyyyyy' received during OPEN | CLOSE processing for Archive Manager database zzzz

Explanation: An error has been received from an Archive Manager open or close request. ‘xx’ gives the return code and ‘yyyyyyyy’ the hexadecimal supplementary reason codes received from Archive Manager. Execution of the utility has been terminated with condition code 12.

Action: Refer to the Archive Manager User Manual for a description of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious reason for the error can be identified, contact your local STK support representative.

▪ OTD140xx – table analysis utility messages

Messages from the DB2 Manager table analysis utility will be written to the SYSPRINT dataset.

OTD14000 Table Analysis Utility ended with completion code xx

Explanation: The DB2 Manager table analysis utility has terminated. ‘xx’ gives the job completion code. This can take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the

SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Action: None. This is an informational message.

OTD14001 OPEN failed for SYSIN|SQLIN dataset

Explanation: An attempt to open the SYSIN or SQLIN file has failed. Execution of the utility will be terminated with condition code 12.

Action: Consult other available system messages for further information on the cause of the open failure. Take the appropriate corrective action and re-run the utility.

OTD14002 The operand for keyword XXXXXXXX in exec parameters is too long

Explanation: The operand specified for the SYSIN parameter keyword ‘xxxxxxxx’ identified in the message is longer than the maximum value permitted. The character string ‘xxxxxxxx’ can take one of the following values: PRINT, DB2SSID, PLAN. Execution of the utility will terminate with condition code 12.

Action: Refer to page 106 for a description of SYSIN parameter formats for the table analysis utility. Correct the parameter error and re-submit the job.

OTD14003 Key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx was not found in DB2 table

Explanation: The column name identified in the message has been specified as a key field in the SQLIN file, but the column does not exist in the DB2 table. Execution of the utility is terminated with condition code 12.

Action: Check that the names of the columns which uniquely index each row in the table have been specified correctly in the SQLIN file, and re-submit the job.

OTD14004 Mandatory exec parameters have not been specified

Explanation: One or more mandatory parameters have not been specified in the SYSIN parameter file. Execution of the utility will be terminated with condition code 12.

Action: Refer to page 106 of this manual for a description of SYSIN parameter formats for the table analysis utility.   Correct the parameter error and re-submit the job.

OTD14005 Invalid or duplicate exec | SQLIN parameter. Please refer to the DB2 Manager User Manual for valid specifications

Explanation: A SYSIN parameter, or SQLIN parameter entry is invalid or has been specified more than once. Execution of the utility is terminated with condition code 12.

Action: Refer to page 99 of this manual for a description of valid SYSIN or SQLIN parameter entries. Correct the error and re-submit the job.

OTD14006 Insufficient storage for workareas

Explanation: An attempt by the utility to allocated virtual storage has failed Execution of the utility will be terminated with condition code 12.

Action: Increase the region size specified for execution of the utility and re-submit the job.

OTD14007 Load failed for program DSNALI | DSNHLI2

Explanation: An attempt to dynamically load the identified program has failed. Execution of the utility is terminated with condition code 12.

Action: This message indicates that DB2 run-time modules were unavailable at time of execution. Ensure that the DB2 run-time load library is available to the executing job, and re-submit the job.

OTD14008 Return code ‘xx’ received during execution of TCBTOKEN | ENQ macro

Explanation: An error has been received attempting to execute the system macro identified in the message. ‘xx’ is the return code received from the macro call. Execution of the utility is terminated with condition code 12.

Action: Consult IBM system macro documentation for a description of the error. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD14009 DB2 CAF error occurred during CONNECT | OPEN | CLOSE | DISCONNECT processing. Return Code = xxx, Reason Code = X'yyyyyyyy’

Explanation: A DB2 Call Attachment Facility error has occurred attempting to perform the type of processing identified in the message. ‘xxx’ gives the return code and ‘yyyyyyyy’ the hexadecimal reason code returned by the Call Attachment Facility. Execution of the utility is terminated with condition code 12.

Action: Refer to the DB2 Application Programming and SQL Guide for a description of CAF error and reason codes. Take the appropriate corrective action and re-start the utility.

OTD14010 SQL Error during execution of xxxxxxxx command SQLCODE = xxxx

Explanation: An error has been received executing the SQL command identified in the message. Processing of the utility is terminated with a condition code 12. All DB2 and Archive Manager updates performed by the utility will be rolled-back to the last commit point. The following SQL commands may appear in this message: DESCRIBE, PREPARE, DECLARE, SET, EXECUTE, OPEN, FETCH, CLOSE, COMMIT. The message will be followed by a line containing parameters from the appropriate SQL error message.

Action: Refer to the DB2 Messages and Codes manual for a description of the SQL error code displayed in the message. Take the appropriate corrective action and re-submit the job. No other special actions are required.

OTD14011 DB2 Table contains Large Objects (LOBs) or 'distinct types’

Explanation: The DB2 table specified in the SQLIN file cannot be processed, because it contains large objects (LOBs) or ‘distinct types’. Tables containing these entities are not eligible for migration using DB2 Manager. Execution of the utility is terminated with a condition code 12.

Action: Check that the correct DB2 table name has been specified in the SQLIN file. If not, correct the entry and re-submit the job.

OTD14012 All columns in the DB2 table have been specified as KEY fields

Explanation: There are no non-indexed columns in the table being analyzed. The utility requires that there be at least one non-index column in the table. Processing of the utility is terminated with condition code 12.

Action: Check that the correct table name has been specified in the SQLIN statement. If it has, and the table contains no non-index columns, then the utility cannot be executed against this table.

OTD14013 Retrieval using the specified KEY fields cannot be made via INDEXONLY processing

Explanation: Scanning of the table using the key fields specified in the SELECT statement in the SQLIN file cannot be performed via index-only processing. Execution of the utility cannot continue, and is terminated with a condition code 12.

Action: Modify the SELECT statement in the SQLIN file to ensure that the specified columns are those used to uniquely index a row in the table, and re-submit the job.

OTD14014 Return Code xx, Supp codes X'yyyyyyyy' received during OPEN | QUERY | CLOSE processing for Archive Manager database zzzz

Explanation: An error has been received from an Archive Manager open, query or close request. ‘xx’ gives the return code and ‘yyyyyyyy’ the hexadecimal supplementary reason codes received from Archive Manager. Execution of the utility has been terminated with condition code 12.

Action: Refer to the Archive Manager User Manual for a description of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious reason for the error can be identified, contact your local STK support representative.

OTD170xx – database housekeeping utility messages

Messages from the DB2 Manager database housekeeping utility will be written to the SYSPRINT.

OTD17000 Database Housekeeping Utility ended with completion code xx

Explanation: The DB2 Manager database housekeeping utility has terminated. 'xx' gives the job completion code. This can take the following values:

0 - the utility has been executed successfully.

4 - a warning condition has been encountered during processing of the utility. A warning message will have been written to the SYSPRINT dataset. The utility has been executed successfully.

8 - an error condition has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has continued.

12 - a serious error has been encountered during processing of the utility. An error message will have been written to the SYSPRINT dataset. Execution of the utility has been terminated.

Action: None. This is an informational message.

OTD17001 OPEN failed for SYSIN|SQLIN dataset

Explanation: An attempt to open the SYSIN or SQLIN file has failed. Execution of the utility will be terminated with condition code 12.

Action: Consult other available system messages for further information on the cause of the open failure. Take the appropriate corrective action and re-run the utility.

OTD17002 The operand for keyword xxxxxxxx in exec parameters is too long

Explanation: The operand specified for the SYSIN parameter keyword 'xxxxxxxx' identified in the message is longer than the maximum value permitted. The character string 'xxxxxxxx' can take one of the following values: PRINT, DB2SSID, PLAN. Execution of the utility will terminate with condition code 12.

Action: Refer to page 111 of this manual for a description of SYSIN parameter formats for the database housekeeping utility. Correct the parameter error and re-submit the job.

OTD17003 Key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx was not found in DB2 table

Explanation: The column name identified in the message has been specified as a key field in the SQLIN file, but the column does not exist in the DB2 table. Execution of the utility is terminated with condition code 12.

Action: Check that the names of the columns which uniquely index each row in the table have been specified correctly in the SQLIN file, and re-submit the job.

OTD17004 Mandatory exec parameters have not been specified

Explanation: One or more mandatory parameters have not been specified in the SYSIN parameter file. Execution of the utility will be terminated with condition code 12.

Action: Refer to page 111 of this manual for a description of SYSIN parameter formats for the database housekeeping utility. Correct the parameter error and re-submit the job.

OTD17005 Invalid or duplicate exec | SQLIN parameter. Please refer to the DB2 Manager User Manual for valid specifications

Explanation: A SYSIN parameter, or SQLIN parameter entry is invalid or has been specified more than once. Execution of the utility is terminated with condition code 12.

Action: Refer to page 109 of this manual for a description of valid SYSIN or SQLIN parameter entries. Correct the error and re-submit the job.

OTD17006 Insufficient storage for workareas

Explanation: An attempt by the utility to allocated virtual storage has failed Execution of the utility will be terminated with condition code 12.

Action: Increase the region size specified for execution of the utility and re-submit the job.

OTD17007 Load failed for program DSNALI | DSNHLI2

Explanation: An attempt to dynamically load the identified program has failed. Execution of the utility is terminated with condition code 12.

Action: This message indicates that DB2 run-time modules were unavailable at time of execution. Ensure that the DB2 run-time load library is available to the executing job, and re-submit the job.

OTD17008 Return code ‘xx’ received during execution of TCBTOKEN | ENQ macro

Explanation: An error has been received attempting to execute the system macro identified in the message. ‘xx’ is the return code received from the macro call. Execution of the utility is terminated with condition code 12.

Action: Consult IBM system macro documentation for a description of the error. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be determined, contact your local STK support representative.

OTD17009 DB2 CAF error occurred during CONNECT | OPEN | CLOSE | DISCONNECT processing. Return Code = xxx, Reason Code = X'yyyyyyyy’

Explanation: A DB2 Call Attachment Facility error has occurred attempting to perform the type of processing identified in the message. ‘xxx’ gives the return code and ‘yyyyyyyy’ the hexadecimal reason code returned by the Call Attachment Facility. Execution of the utility is terminated with condition code 12.

Action: Refer to the DB2 Application Programming and SQL Guide for a description of CAF error and reason codes. Take the appropriate corrective action and re-start the utility.

OTD17010 SQL Error during execution of xxxxxxxx command SQLCODE = xxxx

Explanation: An error has been received executing the SQL command identified in the message. Processing of the utility is terminated with a condition code 12. All DB2 and Archive Manager updates performed by the utility will be rolled-back to the last commit point. The following SQL commands may appear in this message: DESCRIBE, PREPARE, DECLARE, SET, EXECUTE, OPEN, FETCH, CLOSE. The message will be followed by a line containing parameters from the appropriate SQL error message.

Action: Refer to the DB2 Messages and Codes manual for a description of the SQL error code displayed in the message. Take the appropriate corrective action and re-submit the job. No other special actions are required.

OTD17011 DB2 Table contains Large Objects (LOBs) or 'distinct types’

Explanation: The DB2 table specified in the SQLIN file cannot be processed, because it contains large objects (LOBs) or ‘distinct types’. Tables containing these entities are not eligible for migration using DB2 Manager. Execution of the utility is terminated with a condition code 12.

Action: Check that the correct DB2 table name has been specified in the SQLIN file. If not, correct the entry and re-submit the job.

OTD17013 Retrieval using the specified KEY fields cannot be made via INDEXONLY processing

Explanation: Scanning of the table using the key fields specified in the SELECT statement in the SQLIN file cannot be performed via index-only processing. Execution of the utility cannot continue, and is terminated with a condition code 12.

Action: Modify the SELECT statement in the SQLIN file to ensure that the specified columns are those used to uniquely index a row in the table, and re-submit the job.

OTD17014 Return Code xx, Supp codes X'yyyyyyyy' received during OPEN | QUERY | DELETE | CLOSE processing for Archive Manager database zzzz

Explanation: An error has been received from an Archive Manager open, query, delete or close request. 'xx' gives the return code and 'yyyyyyyy' the hexadecimal supplementary reason codes received from Archive Manager. Execution of the utility has been terminated with condition code 12.

Action: Refer to the Archive Manager User Manual for a description of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the job. If no obvious reason for the error can be identified, contact your local STK support representative.

■ OTD200xx – control region master processor messages

Messages from the DB2 Manager control region master processor are written to the system log.

OTD20000 Enter request:

Explanation: This is the operator reply message displayed by the control region after successful initialization.

Action: All DB2 Manager operator commands are entered as responses to this message. After processing a command, the control region will redisplay this message.

OTD20001 Error opening parameter file SYSPARM

Explanation: An error has occurred opening the DB2 Manager parameter library during initialization processing. Initialization is terminated.

Action: Examine other system messages for more information on the open error. Take the appropriate corrective action and restart the control region.

OTD20002 Error X'xxxx' locating member ENVCNTL | TAPECNTL on SYSPARM

Explanation: Hexadecimal error code 'xxxx' has been received attempting to locate the ENVCNTL or TAPECNTL member on the DB2 Manager parameter library. Control region initialization processing is aborted.

Action: Refer to IBM documentation of the LOCATE macro for a description of the error code. Take the appropriate corrective action and restart the control region.

OTD20003 ENVCNTL | TAPECNTL member not found in parameter file

Explanation: The member identified in the message was not present on the DB2 Manager parameter library during control region initialization. Processing is aborted.

Action: Add the missing member to the Archive Manager/OAM parameter library. Refer to chapter 3 for a description of parameter library setup.

OTD20004 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Explanation: This message is used to display a record from an DB2 Manager parameter library member which contains an invalid parameter, and a message explaining the nature of the error.

Two OTD20004 messages are used to display the record - the first message displays characters 1-40 of the record, and the second message displays characters 41-80.

OTD20005 ENVCNTL | TAPECNTL parameters being processed

Explanation: The DB2 Manager parameter library member identified in the message is about to be processed during control region initialization.

Action: None. This is an informational message.

OTD20006 ENVCNTL | TAPECNTL parameter validation errors

Explanation: Errors were detected during validation of parameters in the ENVCNTL or TAPECNTL member of the Archive Manager/OAM parameter library during control region initialization processing. The control region has been terminated.

Action: Refer to the prior OTD20004 messages for details of the individual parameter validation errors. Correct the invalid parameters and restart the control region.

OTD20007 ENVCNTL | TAPECNTL parameters successfully processed

Explanation: Validation of parameters in the ENVCNTL or TAPECNTL member of the DB2 Manager parameter library has completed successfully. Control region initialization processing continues.

Action: None. This is an informational message.

OTD20008 ODCT GETMAIN failure x'xxxx'

Explanation: A request for virtual storage in subpool 241 of the MVS ECSA has been denied by the operating system with return code 'xx'. Initialization of the control region is aborted.

Action: Refer to IBM documentation on the GETMAIN macro for an explanation of the return code. Take the appropriate corrective action and restart the control region.

OTD20009 Communications table initialized

Explanation: The DB2 Manager communications table has been successfully initialized during control region initialization. Processing continues.

Action: None. This is an informational message.

OTD20010 Error X'xx' during initialization of task *tnnn*

Explanation: Error code 'xx' was returned by a sub-task during Archive Manager/OAM initialization processing. Control region initialization has been aborted. ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.

Action: Consult other messages displayed during control region initialization for a description of the sub-task error. Take the action specified for the error message, and restart the control region.

If the cause of the error is not apparent from any displayed message, contact your DB2 Manager product support representative.

OTD20011 Timeout error during initialization of task *tnnn*

Explanation: A control region sub-task has not responded with one minute during initialization processing. Initialization of the control region has been aborted. ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.

Action: Consult other messages displayed during control region initialization for possible information as to why the sub-task is not responding. Take the action specified for the error message, and restart the control region.

If no other abnormal message is present, or the cause of the error is not apparent from any displayed message, contact your DB2 Manager product support representative.

OTD20012 Return code X'xx' retrieving | creating | deleting Name/Token pair

Explanation: An error has occurred retrieving, creating or deleting an OS/390 name/token pair. 'xx' gives the hexadecimal reason code received from the retrieve request. Execution of the utility is terminated with condition code 12.

Action: Refer to IBM documentation on the IEANTRT callable service (for retrieval errors), the IEANTCR callable service (for creation errors) or the IEANTDL callable service (for deletion errors) for a description of the error code. Take the appropriate corrective action and re-submit the job. If no obvious cause for the error can be identified, contact your local STK support representative.

OTD20013 GETMAIN failure X'xx' allocating task *tnnn*

Explanation: A request for virtual storage in the private area of the control region during sub-task creation has been denied by the operating system with return code 'xx'. Initialization of the control region is aborted. 't' identifies the sub-task type ('D' – disk reader; 'H' – housekeeping; 'S' – scheduler; 'T' – tape reader; 'W' – writer) and 'nnn' gives the sub-task number.

Action: Refer to IBM documentation on the GETMAIN macro for an explanation of the return code. Take the appropriate corrective action and restart the control region.

OTD20014 Error X'xx' received attaching task *tnnn*

Explanation: Hexadecimal error code 'xx' was received from the ATTACH macro that was used to initiate a sub-task during DB2 Manager control region initialization processing. The control region is terminated. 't' identifies the sub-task type ('D' – disk reader; 'H' – housekeeping; 'S' – scheduler; 'T' – tape reader; 'W' – writer) and 'nnn' gives the sub-task number.

Action: Refer to IBM documentation on the ATTACH macro for a description of the error code. Take the appropriate corrective action and restart the control region.

If no apparent cause for the failure can be identified, contact your DB2 Manager product support representative.

OTD20015 nnn Scheduler | Tape reader | Disk reader | Tape writer | Housekeeping tasks initialized successfully

Explanation: All sub-tasks of the type identified in the message have been successfully initialized during start-up of the DB2 Manager control region. ‘nnn’ gives the number of sub-tasks initialized.

Action: None. This is an informational message.

OTD20016 OTDBP300 is not in Link Pack Area

Explanation: The DB2 Manager row edit routine OTDBP300 was not found in the FLPA or MLPA during control region initialization processing. The control region is terminated.

Action: Ensure that module OTDBP300 is installed in the FLPA or MLPA before restarting the control region (see “Perform MVS host system modifications” on page21 for a discussion of this requirement).

OTD20017 Intercept successfully established

Explanation: The Archive Manager /DB2 row edit routine OTDBP300 has been located in the FLPA or MLPA, and has been successfully initialized.

Action: None. This is an informational message.

OTD20018 ENQueue Error: Major: mmmmmmmm Minor: nnnnnnnnnnnnnnn Return code=X'xx'

Explanation: An error has been received attempting to enqueue on the resource whose major and minor names are identified in the message. ‘xx’ is the return code received from the ENQ macro.

Action: Refer to IBM documentation on ENQ macro processing for a description of the return code. Take the appropriate corrective action and re-start the control region. If no apparent cause for the error can be detected, contact your DB2 Manager product support representative.

OTD20019 Error locating DB2 Manager ASCB | Jobname

Explanation: An error has been detected attempting to locate the ASCB for the DB2 Manager control region, or retrieving the started task name assigned to the control region.

Action: This indicates an internal error in DB2 Manager processing. Report the error to your local DB2 Manager product support representative.

OTD20020 Lifecycle Director V1.1 initialization starting

Explanation: This message is displayed at start of control region initialization processing.

Action: None. This is an informational message.

OTD20021 Initialization completed successfully

Explanation: All control region initialization processing has been performed successfully.

Action: None. This is an informational message. If COMMAND=REPLY has been specified in the DB2 Manager parameter library member TAPECNTL, it will be followed by message OTM20000.

OTD20025 Error xx Reason xxxxxxxx received during authorization code checking

Explanation: An internal error has occurred during authorization code checking. Control region initialization processing is aborted.

Action: This is an internal error. Report the problem to your DB2 Manager product support representative.

OTD20026 Control region already initialized - job terminating

Explanation: An attempt has been made to start the DB2 Manager control region, but a control region is already active in the same system image. Initialization of the second control region is terminated.

Action: Ensure that any existing control region execution is terminated before attempting to start the control region.

OTD20027 System abend 'xxx' intercepted – processing will continue

Explanation: A system abend has been intercepted during control region operation. The system abend code is 'xxx'. Control region processing will continue.

Action: A system abend has been trapped during processing of an operator request. If possible, DB2 Manager will recover from the abend. Operator requests may continue to be entered as normal.

If it is not possible for DB2 Manager to recover from the abend, control region processing will be terminated abnormally.

The problem should be reported to your DB2 Manager product support representative.

OTD20028 Unable to continue after abend - job terminating

Explanation: The DB2 Manager control region was unable to recover after trapping an abend condition. The control region will be terminated abnormally.

Action: Report the problem to your DB2 Manager product support representative. Preserve all diagnostic information and system dumps for further analysis.

OTD20029 Recursive abends intercepted - job terminating

Explanation: A recursive abend has occurred after the DB2 Manager control region trapped a system abend condition. The control region will be terminated abnormally.

Action: Report the problem to your DB2 Manager product support representative. Preserve all diagnostic information and system dumps for further analysis.

OTD20030 Error received posting requestor ECB

Explanation: An error occurred posting an outstanding request for access to a migrated row. The DB2 Manager control region will attempt to recover from this error condition and continue processing.

Action: Consult other available system messages for additional information regarding the error. If no apparent cause for the failure can be determined, report the problem to your DB2 Manager product support representative. Preserve all diagnostic information and system dumps for further analysis.

OTD20031 Initialization processing aborted

Explanation: An error has occurred during DB2 Manager control region initialization processing. Initialization has been terminated.

Action: Examine other messages displayed by the control region during the initialization process for details of the error. Take the appropriate corrective action and restart the control region.

OTD20032 Command rejected - enter request via MODIFY command

Explanation: ‘COMMAND=MODIFY’ is set in the Archive Manager/OAM TAPECNTL parameter library member, but an MVS command other than ‘MODIFY’ has been issued to the DB2 Manager control region.

Action: Re-enter the DB2 Manager command using the standard MVS MODIFY command structure. Refer to “DB2 Manager operator interface” on page 50 for more information on entering DB2 Manager control region commands.

OTD20033 I/O error X'xxxx' processing ENVCNTL | TAPECNTL parameter member

Explanation: An I/O error has occurred processing the identified member from the DB2 Manager parameter library during control region initialization processing. 'xxxx' gives the first two sense bytes from the IOB at the time of the error. Control region initialization is aborted.

Action: Refer to IBM documentation for a description of the error code. Take the appropriate corrective action and restart the control region.

If no apparent cause for the error can be detected, contact your DB2 Manager product support representative.

OTD20034 Error X’xx’ received loading program XXXXXXXX

Explanation: An error has occurred dynamically loading the program identified in the message. ‘xx’ gives the hexadecimal error record received from the LOAD request.

Action: Refer to IBM macro documentation for a description of the error code in the message.   Ensure that the program identified in the message is available to be loaded from the steplib, joblib or linklist concatenation. Consult other available system messages for additional information on the load error. Take the appropriate corrective action and restart the control region.

OTD20049 Task *tnnn* shutdown request has failed - error X'xx'

Explanation: Shutdown of a sub-task has failed during control region termination processing. ‘xx’ gives the hexadecimal return code from the DETACH macro. ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.   Termination of the control region continues.

Action: None. This is an informational message.   System abend ‘A03’ may be received on termination of the control region after display of this message.

OTD20050 Missing or invalid request - ignored

Explanation: An invalid operator command has been entered in response to the DB2 Manager OTD20000 message or via MODIFY command.

Action: Re-enter the correct operator command in response to the OTD20000 message or via MODIFY command.

OTD20051 Invalid Set operand(s) - request ignored

Explanation: An invalid operand has been entered with the SET operator command.

Action: Ensure that the SET command is specified correctly and re-enter the command. Refer to page 58 for details of the SET operator command format.

OTD20052 Configuration altered successfully

Explanation: The operator SET command has been processed successfully to alter the current DB2 Manager configuration.

Action: None. This is an informational message. If COMMAND=REPLY has been specified in the DB2 Manager TAPECNTL parameter library member, the OTD20000 operator reply message will be redisplayed.

OTD20053 Unable to reduce value of MAXTRDR | MAXDRDR | MAXTWTR | MAXSCHED as requested

Explanation: All available tasks have been examined but the control region has been unable to reduce the value of the initialization parameter specified in the message to that requested.

Action: This is an DB2 Manager error condition. Report the message to your DB2 Manager product support representative. Obtain as much information as you can about the status of the DB2 Manager control region at the time of the message display.

OTD20054 Timeout error occurred when acquiring the Scheduler lock

Explanation: The control region has been unable to gain exclusive access to the scheduler lock within 1 minute. Processing of the operator command is unable to continue.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD20055 Missing/invalid operand - request rejected

Explanation: A mandatory operand is missing from an DB2 Manager operator command.

Action: Re-enter the command with the correct syntax. Refer to “DB2 Manager operator interface” on page 50 for a description of operator command formats.

OTD20056 Errors altering configuration – check system status

Explanation: Errors have been detected attempting to alter the current DB2 Manager configuration using the operator SET command. Other messages will give details of the error encountered.

Action: Check the system status via the operator DISPLAY command to verify that the system configuration has not been altered erroneously.

OTD20057 Requested task *tnnn* does not exist - request ignored

Explanation: A sub-task specified in an operator command does not exist. The request is ignored. ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.

Action: Ensure that the correct task number is specified in the operator command. If necessary, use the DISPLAY command to display details of all currently active sub-tasks.

OTD20058 Display processing completed

Explanation: Processing of an operator DISPLAY command has been successfully completed.

Action: None. This is an informational message.

OTD20059 Task *tnnn* is not active - request ignored

Explanation: A request to purge a sub-task in the control region cannot be performed because the task is either not initialized or already terminated. ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.

Action: Ensure that the correct task number is specified in the operator command. If necessary, use the DISPLAY command to display details of all currently active sub-tasks.

OTD20060 Task *tnnn* has been shutdown successfully

Explanation: A request to purge the identified sub-task has been completed successfully. ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.

Action: None. This is an informational message.

OTD20061 Invalid Display operand(s) – request ignored

Explanation: An invalid operand has been entered with the DISPLAY operator command.

Action: Ensure that the DISPLAY command is specified correctly and re-enter the command. Refer to page 54 for a specification of the DISPLAY operator command.

OTD20062 Maxtrdr=nnn Maxdrdr=nnn Maxtwtr=nnn Maxsched=nnn Maxqlen=nnn

OTD20062 Retaintape= nnnn Tapewait=nnnn Objsize=nnnnn

OTD20062 Readtimeout=nnnn Writetimeout=nnnn

OTD20062 Scheduler qlen=nnn unlocked | lkd/Xnnn; held request qlen=nnn

Explanation: This group of messages is displayed in response to an operator DISPLAY command, and gives a summary of current control region configuration settings.

For a locked scheduler queue display, ‘Xnnn’ identifies the lock owner, where ‘X’ is the resource type (‘M’ = master task, ‘S’ = scheduler task, ‘P’ = external program), and ‘nnn’ is the resource number.

Action: None. This is an information message only.

OTD20063 Task *tnnn*: Status=aaaa Qlen=bbb DBid=cccc Stg_level=d Tape=eeeee Lastuse=hh:mm:ss

Explanation: This message is displayed in response to a detailed or summary operator DISPLAY command. For summary commands, it will be displayed once for each subtask in the control region, where:

‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number.

aaaa’ indicates the task status, and can take the following values:

NULL – the task has not yet been initialized

WAIT – the task is currently waiting for work
WTSH – a shutdown request has been queued for the task

BUSY – the task is currently processing requests
BYSH – the task is currently processing requests, and will terminate when all requests have been processed
TERM – the task has terminated

’bbb’ gives the number of requests currently queued for the task. For busy reader and writer tasks, ‘cccc’ gives the identifier of the database being processed by the subtask. For busy reader tasks or waiting tape reader tasks which have retained allocation of a tape volume, ‘d’ gives the database storage level and ‘eeeee’ the sequence number within storage level of the tape or disk dataset being processed. The ‘Lastuse’ entry will only be displayed for waiting tape reader tasks which have retained allocation of a tape volume. ‘hh:mm:ss gives the time the tape was last in use. The tape volume will be automatically de-allocated and dismounted if a retrieval request for that volume is not received within the time specified in the RETAINTAPE system initialization parameter.

Action: None. This is an informational message.

OTD20066 Request nnn: Type=xxxxxxxx DBid=yyyy Jobname=zzzzzzzz

Explanation: This message is displayed for busy tasks in response to a detailed operator DISPLAY command. ‘nnn’ gives the position of the request in the sub-task request queue. ‘xxxxxxxx’ gives the type of request, and can take the following values: ‘Open’, ‘Retrieve’, ‘Store’, ‘Commit’ or ‘Close’. ‘yyyy’ gives the identifier of the Archive Manager database being processed by the subtask, and ‘zzzzzzzz’ the name of the job from which the request was originally submitted.

Action: None. This is an informational message.

OTD20067 Request nnn: Stg Level=x Tape=yyyyy Blockid=zzzzzzzz

Explanation: This message is displayed as a supplement to message OTD20066 for retrieve or store requests. ‘nnn’ gives the position of the request in the sub-task request queue, ‘x’ the Archive Manager database storage level and ‘yyyyy’ the sequence number within the database storage level of the disk or tape dataset being processed by the sub-task. ‘zzzzzzzz’ gives the hexadecimal identifier of the block containing the migrated row being processed by the sub-task.

Action: None. This is an informational message only.

OTD20071 Invalid Purge operand(s) – request ignored

Explanation: An invalid operand has been entered with the PURGE operator command.

Action: Ensure that the PURGE command is specified correctly and re-enter the command. Refer to page 55 for details of the operator PURGE command.

OTD20072 Task *tnnn* is already being purged – request ignored

Explanation: A second PURGE request has been entered for a busy sub-task before the first PURGE request has been processed, where 't' identifies the sub-task type ('D' – disk reader; 'H' – housekeeping; 'S' – scheduler; 'T' – tape reader; 'W' – writer) and 'nnn' gives the sub-task number.. The request is ignored.

Action: Wait for the queued PURGE command to be processed after all outstanding retrieval requests have been completed. To immediately terminate the sub-task, enter the FORCE command.

OTD20073 Task *tnnn* purge request is being processed

Explanation: A PURGE request has been queued for a sub-task, where 't' identifies the sub-task type ('D' – disk reader; 'H' – housekeeping; 'S' – scheduler; 'T' – tape reader; 'W' – writer) and 'nnn' gives the sub-task number. Waiting tasks will terminate immediately; busy tasks will terminate when all currently queued requests have been processed.

Action: None this is an informational message.

OTD20081 Invalid Force operand(s) – request ignored

Explanation: An invalid operand has been entered with the FORCE operator command.

Action: Ensure that the FORCE command is specified correctly and re-enter the command. Refer to section 0 on page 4.55 for details of the FORCE operator command format.

OTD20082 Controller received error posting requestor ECB

Explanation: An error occurred posting an outstanding request during force purge processing. The DB2 Manager control region will attempt to recover from this error condition and continue processing.

Action: Consult other available system messages for additional information regarding the error. If no apparent cause for the failure can be determined, report the problem to your DB2 Manager product support representative. Preserve all diagnostic information and system dumps for further analysis.

OTD20083 Task *tnnn* has not been purged – request ignored

Explanation: A FORCE request has been entered for task 'tnnn', where ‘t’ identifies the sub-task type (‘D’ – disk reader; ‘H’ – housekeeping; ‘S’ – scheduler; ‘T’ – tape reader; ‘W’ – writer) and ‘nnn’ gives the sub-task number, but no prior PURGE request has been entered. A PURGE request must already have been entered for a task before a FORCE request is accepted. The request is ignored.

Action: Enter a PURGE request for the task. All outstanding retrieval requests will be completed before the specified task is terminated. If necessary a FORCE request may then be entered for the task.

OTD20091 Invalid Shutdown operand(s)

Explanation: An invalid operand has been entered with the operator SHUTDOWN command.

Action: Re-enter the command, specifying a valid operand. Refer to page 62 for details of the operator SHUTDOWN command format.

OTD20092 Do you want to proceed with shutdown (y/n)?

Explanation: An operator SHUTDOWN request has been entered on the console via the operator reply facility. This message is displayed to ask for confirmation of the shutdown request.

Action: To proceed with DB2 Manager control region shutdown processing, enter 'Y'.

To abort the shutdown request, enter 'N'. The operator reply message OTD20000 will then be redisplayed.

OTD20093 Shutdown in progress

Explanation: An operator SHUTDOWN request has been entered and accepted. Termination of the control region will continue.

Action: None. This is an informational message only.

OTD20094 Error freeing communications table storage

Explanation: The storage area allocated in subpool 241 in the MVS ECSA during control region initialization processing for the DB2 Manager communications table was not freed successfully during control region shutdown. Termination processing continues.

Action: Refer to other available system messages for more information regarding the error. When the cause has been identified, take the appropriate corrective action to ensure that the error does not re-occur during future shutdown of the control region.

If no apparent cause of the error can be identified, contact your DB2 Manager product support representative.

The ECSA storage used for the DB2 Manager communications table in this invocation of the control region will be unavailable for allocation until after the next IPL of the system. Approximately 200 kbytes of storage are used for the communications table in extended CSA storage.

OTD20095 Communications table storage area freed

Explanation: The storage area obtained from subpool 241 in extended CSA during control region initialization processing has been successfully released during control region shutdown. Processing continues.

Action: None. This is an informational message.

OTD20096 Shutdown processing complete

Explanation: DB2 Manager shutdown processing has been completed successfully. The control region will terminate normally.

Action: None. This is an informational message.

■ OTD220xx – control region scheduler messages

Messages from the DB2 Manager control region scheduler are written to the system log.

OTD22001 Scheduler initialization error: invalid control block

Explanation: An invalid control block has been encountered by the scheduler task during initialization processing. Control region initialization will be terminated.

Action: This condition indicates an internal error in DB2 Manager processing. Retain as much information about the error as possible and report the problem to your local DB2 Manager support representative.

OTD22002 Scheduler initialization error: insufficient storage available for workareas

Explanation: A request for virtual storage in the DB2 Manager control region has been rejected by the operating system. Control region initialization processing is aborted.

Action: Allocate more virtual storage to the control region, by increasing the value of the SIZE parameter in the DB2 Manager procedure. Restart the control region.

OTD22003 Storage Release Error: Return Code=X'xx' T=Snnn

Explanation: An error has been received trying to free a virtual storage area during scheduler task shutdown processing. ‘xx’ gives the hexadecimal return code from the STORAGE macro, and ‘nnn’ identifies the scheduler task number. Shutdown processing will continue normally.

Action: Consult IBM documentation on the STORAGE macro for an explanation of the return code displayed in the message. If necessary, take appropriate action to ensure the problem does not re-occur. If the error persists, and no apparent cause can be identifier, report the problem to your local DB2 Manager support representative.

OTD22004 Invalid request on Scheduler Queue

Explanation: A request in the scheduler queue is not a valid DB2 Manager request element. Processing of the request cannot continue. The scheduler task will continue by waiting for the next request to arrive.

Action: This message indicates an internal error in DB2 Manager processing. Retain as much information about the error as possible and report the problem to your local DB2 Manager support representative.

OTD22005 ENQueue Error: Major: mmmmmmmm Minor: nnnnnnnnnnnnnnn Return code=X'xx' T=Snnn

Explanation: An error has been received attempting to enqueue on the resource whose major and minor names are identified in the message. ‘xx’ is the return code received from the ENQ macro, and ‘nnn’ identifies the scheduler task receiving the error.

Action: Refer to IBM documentation on ENQ macro processing for a description of the return code. Take the appropriate corrective action and re-start the control region. If no apparent cause for the error can be detected, contact your DB2 Manager product support representative.

OTD22006 Scheduler terminating because of recursive abends T=Snnn

Explanation: An abend condition has been detected by a scheduler task, when attempting to recover from an earlier abend condition. 'nnn' identifies the scheduler task receiving the error. The scheduler task will terminate automatically and the current value of the MAXSCHED initialization parameter will be reduced by 1.

Action: Examine the original abend condition, and take any corrective action required. Other scheduler tasks may continue to process requests normally. To restart the terminated scheduler task, reset the value of MAXSCHED back to its value prior to termination of the scheduler task.

OTD22007 Scheduler received error posting requestor ECB T=Snnn

Explanation: An error occurred posting an outstanding request. The scheduler task will attempt to recover from this error condition and continue processing.

Action: Consult other available system messages for additional information regarding the error. If no apparent cause for the failure can be determined, report the problem to your DB2 Manager product support representative. Preserve all diagnostic information and system dumps for further analysis.

OTD22008 Scheduler task nnn abend Sxxx intercepted

Explanation: A system abend has been intercepted during operation of scheduler task 'nnn'. The system abend code is 'xxx'. Scheduler processing will continue.

Action: A system abend has been trapped during processing of a migrated row access request. If possible, DB2 Manager will recover from the abend.

If it is not possible for DB2 Manager to recover from the abend, scheduler task processing will be terminated abnormally.

The problem should be reported to your DB2 Manager product support representative.

OTD22009 Scheduler task nnn terminated

Explanation: Scheduler task number 'nnn' has terminated processing in response to an operator command, or following an unrecoverable error condition.

Action: None. This is an informational message.

OTD22011 VSAM error X'aabb' received opening | reading dataset OTASxxxx, T=Snnn'

Explanation: An error has been received by scheduler task ‘nnn’ opening or reading the primary index dataset of Archive Manager database ‘xxxx’. ‘aa’ and ‘bb’ give the hexadecimal return and reason codes from the VSAM request. The migrated row access request being processed by the task will be rejected.

Action: Refer to IBM VSAM macro documentation for an explanation of the return and reason codes displayed in the message. Take the appropriate corrective action before re-submitting the request which generated the error condition.

OTD22040 System abend ‘xxx’ intercepted – control region will attempt to recover

Explanation: A system abend condition has occurred during scheduler operation, and has been intercepted by the program. The scheduler will attempt to recover from this condition and continue processing normally.

Action: Refer to IBM documentation for an explanation of the system abend code displayed in the message. If no apparent cause for the problem can be identifier, report the error to your StorageTek customer support representative.

■ OTD250xx – control region tape reader messages

Messages from DB2 Manager control region tape reader tasks are written to the system log.

OTD25001 Invalid ODRH received on entry

Explanation: An invalid reader task header control block has been encountered by a tape reader task during initialization processing. Reader task initialization will be terminated.

Action: This condition indicates an internal error in DB2 Manager processing. Retain as much information about the error as possible and report the problem to your local DB2 Manager support representative.

OTD25002 Unable to get exclusive access to TAPEREADER resource

Explanation: A tape reader task has been unable to gain exclusive access to the tape reader request queue within 1 minute. Processing of the request continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD25003 Unable to get exclusive access to ODRE resource

Explanation: A tape reader task has been unable to gain exclusive access to the scheduler request queue within 1 minute. Processing of the request continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD25004 Insufficient storage for workareas

Explanation: A request by a reader task for virtual storage in the DB2 Manager control region has been rejected by the operating system. Tape reader task initialization processing is aborted.

Action: Allocate more virtual storage to the control region, by increasing the value of the SIZE parameter in the DB2 Manager procedure. Restart the control region.

OTD25005 Tape rdr task nnn initialization failed

Explanation: Initialization of tape reader sub-task ‘nnn’ has failed. Tape reader task initialization processing will be terminated.

Action: Examine the system log for other messages displayed by the DB2 Manager, giving the cause of tape reader task initialization failure. Take the appropriate action described for those messages, and retry the tape reader task initialization request.

OTD25007 Tape rdr task nnn abend Sxxx intercepted

Explanation: A system abend has been intercepted during operation of tape reader task ‘nnn’. The system abend code is 'xxx'. Tape reader processing will continue.

Action: A system abend has been trapped during processing of a migrated row access request. If possible, DB2 Manager will recover from the abend.

If it is not possible for DB2 Manager to recover from the abend, the tape reader task will be terminated abnormally.

The problem should be reported to your DB2 Manager product support representative.

OTD25009 Tape rdr task nnn terminated

Explanation: Tape reader task number ‘nnn’ has terminated processing in response to an operator command, or following an unrecoverable error condition.

Action: None. This is an informational message.

OTD25011 Return code xx Supp codes X'yyyyzzzz' received opening | reading | closing d/b dddd, T=Tnnn

Explanation: An error has been received by tape reader task ‘nnn’ accessing the Archive Manager database with identifier ‘dddd’, for the type of processing identified in the message. ‘xx’ gives the return code received from the Archive Manager call, and ‘yyyy’ and ‘zzzz’ give supplementary error codes 1 and 2. Processing of the current request is terminated with an error.

Action: Refer to the Archive Manager User Manual for an explanation of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the request.

OTD25012 Unrecoverable error condition - tape reader task nnn terminating

Explanation: An error condition has been encountered during processing of a request by tape reader task ‘nnn’, and the task is unable to recover. The request will be terminated with an error condition and the reader task will terminate. The value of the MAXTRDR initialization parameter will be automatically decremented by 1.

Action: Examine the system log for other DB2 Manager messages giving information about the original error condition. Take the appropriate corrective action and re-submit the failing request. To restart the terminated reader task, set the value of MAXTRDR to its value prior to termination of the task.

OTD25013 Error xx received writing SMF record

Explanation: An error has been detected by a tape reader task writing a record to the SMF dataset. ‘xx’ gives the return code received from the SMFWTM macro. Tape reader task processing will continue normally.

Action: Refer to IBM documentation for an explanation of the return code displayed in the message. Take the appropriate corrective action to resolve the problem.

OTD25040 System abend ‘xxx’ intercepted – control region will attempt to recover

Explanation: A system abend condition has occurred during tape reader task operation, and has been intercepted by the program. The tape reader task will attempt to recover from this condition and continue processing normally.

Action: Refer to IBM documentation for an explanation of the system abend code displayed in the message. If no apparent cause for the problem can be identifier, report the error to your StorageTek customer support representative.

OTD254xx – control region disk reader messages

Messages from DB2 Manager control region disk reader tasks are written to the system log.

OTD25401 Invalid ODRH received on entry

Explanation: An invalid reader task header control block has been encountered by a disk reader task during initialization processing. Reader task initialization will be terminated.

Action: This condition indicates an internal error in DB2 Manager processing. Retain as much information about the error as possible and report the problem to your local DB2 Manager support representative.

OTD25402 Unable to get exclusive access to DISKREADER resource

Explanation: A disk reader task has been unable to gain exclusive access to the disk reader request queue within 1 minute. Processing of the request continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD25403 Unable to get exclusive access to ODRE resource

Explanation: A disk reader task has been unable to gain exclusive access to the scheduler request queue within 1 minute. Processing of the request continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD25404 Insufficient storage for workareas

Explanation: A request by a disk reader task for virtual storage in the DB2 Manager control region has been rejected by the operating system. Disk reader task initialization processing is aborted.

Action: Allocate more virtual storage to the control region, by increasing the value of the SIZE parameter in the DB2 Manager procedure. Restart the control region.

OTD25405 Disk rdr task nnn initialization failed

Explanation: Initialization of disk reader sub-task 'nnn' has failed. Disk reader task initialization processing will be terminated.

Action: Examine the system log for other messages displayed by the DB2 Manager, giving the cause of tape reader task initialization failure. Take the appropriate action described for those messages, and retry the disk reader task initialization request.

OTD25407 Disk rdr task nnn abend Sxxx intercepted

Explanation: A system abend has been intercepted during operation of disk reader task 'nnn'. The system abend code is 'xxx'. Disk reader processing will continue.

Action: A system abend has been trapped during processing of a migrated row access request. If possible, DB2 Manager will recover from the abend.

If it is not possible for DB2 Manager to recover from the abend, the disk reader task will be terminated abnormally.

The problem should be reported to your DB2 Manager product support representative.

OTD25409 Disk rdr task nnn terminated

Explanation: Disk reader task number 'nnn' has terminated processing in response to an operator command, or following an unrecoverable error condition.

Action: None. This is an informational message.

OTD25411 Return code xx Supp codes X'xxxxxxxx' received opening | reading | closing d/b dddd, T=Dnnn

Explanation: An error has been received by disk reader task 'nnn' accessing the Archive Manager database with identifier 'dddd', for the type of processing identified in the message. 'xx' gives the return code received from the Archive Manager call, and 'yyyy' and 'zzzz' give supplementary error codes 1 and 2. Processing of the current request is terminated with an error.

Action: Refer to the Archive Manager User Manual for an explanation of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-submit the request.

OTD25412 Unrecoverable error condition - disk reader task nnn terminating

Explanation: An error condition has been encountered during processing of a request by disk reader task 'nnn', and the task is unable to recover. The request will be terminated with an error condition and the reader task will terminate. The value of the MAXDRDR initialization parameter will be automatically decremented by 1.

Action: Examine the system log for other DB2 Manager messages giving information about the original error condition. Take the appropriate corrective action and re-submit the failing request. To restart the terminated reader task, set the value of MAXDRDR to its value prior to termination of the task.

OTD25413 Error xx received writing SMF record

Explanation: An error has been detected by a disk reader task writing a record to the SMF dataset. 'xx' gives the return code received from the SMFWTM macro. Disk reader task processing will continue normally.

Action: Refer to IBM documentation for an explanation of the return code displayed in the message. Take the appropriate corrective action to resolve the problem.

■ OTD260xx – control region writer messages

Messages from DB2 Manager control region writer tasks are written to the system log.

OTD26001 Invalid ODWH received on entry

Explanation: An invalid writer task header control block has been encountered by a tape reader task during initialization processing. Writer task initialization will be terminated.

Action: This condition indicates an internal error in DB2 Manager processing. Retain as much information about the error as possible and report the problem to your local DB2 Manager support representative.

OTD26002 Unable to get exclusive access to TAPEWRITER resource

Explanation: A tape writer task has been unable to gain exclusive access to the tape writer request queue within 1 minute. Processing of the request continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD26003 Unable to get exclusive access to ODRE resource

Explanation: A tape writer task has been unable to gain exclusive access to the scheduler request queue within 1 minute. Processing of the request continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD26004 Insufficient storage for workareas

Explanation: A request by a writer task for virtual storage in the DB2 Manager control region has been rejected by the operating system. Tape writer task initialization processing is aborted.

Action: Allocate more virtual storage to the control region, by increasing the value of the SIZE parameter in the DB2 Manager procedure. Restart the control region.

OTD26005 Tape wtr task nnn initialization failed

Explanation: Initialization of tape writer sub-task ‘nnn’ has failed. Tape writer task initialization processing will be terminated.

Action: Examine the system log for other messages displayed by the DB2 Manager, giving the cause of tape writer task initialization failure. Take the appropriate action described for those messages, and retry the tape writer task initialization request.

OTD26007 Tape wtr task nnn abend Sxxx intercepted

Explanation: A system abend has been intercepted during operation of tape writer task ‘nnn’. The system abend code is 'xxx'. Tape writer processing will continue.

Action: A system abend has been trapped during processing of a migrated row storage request. If possible, DB2 Manager will recover from the abend.

If it is not possible for DB2 Manager to recover from the abend, the tape writer task will be terminated abnormally.

The problem should be reported to your DB2 Manager product support representative.

OTD26009 Tape wtr task nnn terminated

Explanation: Tape writer task number ‘nnn’ has terminated processing in response to an operator command, or following an unrecoverable error condition.

Action: None. This is an informational message.

OTD26011 Return code xx Supp codes X'yyyyzzzz' received opening | updating | committing | closing d/b dddd, T=Wnnn

Explanation: An error has been received by tape writer task ‘nnn’ accessing the Archive Manager database with identifier ‘dddd’, for the type of processing identified in the message. ‘xx’ gives the return code received from the Archive Manager call, and ‘yyyy’ and ‘zzzz’ give supplementary error codes 1 and 2. Processing of the current request is terminated with an error.

Action: Refer to the Archive Manager User Manual for an explanation of the return and supplementary codes displayed in the message. Take the appropriate corrective action and re-start the row migration job.

OTD26012 Unrecoverable error condition – tape writer task nnn terminating

Explanation: An error condition has been encountered during processing of a request by tape writer task ‘nnn’, and the task is unable to recover. The request will be terminated with an error condition and the writer task will terminate. The value of the MAXTWTR initialization parameter will be automatically decremented by 1.

Action: Examine the system log for other DB2 Manager messages giving information about the original error condition. Take the appropriate corrective action and re-start the row migration job. To restart the terminated writer task, set the value of MAXTWTR to its value prior to termination of the task.

OTD26013 Error xx received writing SMF record

Explanation: An error has been detected by a tape writer task writing a record to the SMF dataset. ‘xx’ gives the return code received from the SMFWTM macro. Tape writer task processing will continue normally.

Action: Refer to IBM documentation for an explanation of the return code displayed in the message. Take the appropriate corrective action to resolve the problem.

OTD270xx - control region housekeeping task messages

Messages from the DB2 Manager control region housekeeping tasks are written to the system log.

OTD27001 Invalid ODCT received on entry

Explanation: An invalid communications table control block has been encountered by the housekeeping task during initialization processing. Housekeeping task initialization will be terminated.

Action: This condition indicates an internal error in DB2 Manager processing. Retain as much information about the error as possible and report the problem to your local DB2 Manager support representative.

OTD27002 Unable to get exclusive access to ODRE resource

Explanation: A housekeeping task has been unable to gain exclusive access to the scheduler request queue within 1 minute. Processing continues, but request queue integrity in the DB2 Manager control region may have been compromised.

Action: Examine the system log for other abnormal messages issued by the DB2 Manager control region. If there is no apparent reason for the failure, contact your local DB2 Manager product support representative.

OTD27003 ODRE table is not valid

Explanation: The table of DB2 Manager request elements in extended common storage is not in a valid format. Housekeeping processing cannot continue, and the task will terminate.

Action: This indicates an internal error in DB2 Manager processing. Report the problem to your local DB2 Manager support representative.

OTD27004 Insufficient storage for workareas

Explanation: A request by a housekeeping task for virtual storage in the DB2 Manager control region has been rejected by the operating system. Housekeeping task initialization processing is aborted.

Action: Allocate more virtual storage to the control region, by increasing the value of the SIZE parameter in the DB2 Manager procedure. Restart the control region.

OTD27005 Housekeeping task Hnnn terminated

Explanation: Housekeeping task number ‘nnn’ has terminated processing in response to an operator command, or following an unrecoverable error condition.

Action: None. This is an informational message.

OTD27006 Unrecoverable error condition - housekeeping task nnn terminating

Explanation: An error condition has been encountered during processing of housekeeping task ‘nnn’, and the task is unable to recover. The housekeeping task will terminate.

Action: Examine the system log for other DB2 Manager messages giving information about the original error condition. Take the appropriate corrective action and re-start the control region.

OTD27007 Housekeeping task nnn abend Sxxx intercepted

Explanation: A system abend has been intercepted during operation of housekeeping task ‘nnn’. The system abend code is 'xxx'. Housekeeping processing will continue.

Action: If possible, DB2 Manager will recover from the abend. If it is not possible for DB2 Manager to recover from the abend, the housekeeping task will be terminated abnormally.

The problem should be reported to your DB2 Manager product support representative.

OTD27008 Hnnn request timed out - ODRE xxxxxxxx

Explanation: A stalled request has been timed out by a housekeeping task. ‘xxxxxxxx’ gives the virtual storage address of the DB2 Manager request element used by the timed out request.

Action: None. This is an informational message.

■ DB2 Manager error and reason codes

Any error condition encountered by DB2 Manager during processing of a row access request generated by an application program SQL command will be communicated back to the calling program via generation of an SQL –652 error. The two bytes at character positions 6 and 7 in field SQLERRD in the SQL communication area (SQLCA) at the time of the error will contain a reason code giving additional information about the error condition. Messages giving additional error information may also be displayed in the system log by the DB2 Manager control region.

This section lists reason codes set by DB2 Manager in the SQLCA. Note that reason codes in this list are displayed in hexadecimal format.

Reason code (hex) ***Explanation***

01xx Storage obtain request by the DB2 Manager row edit processor failed. 'xx' gives the hexadecimal return code received from the STORAGE macro request.

Action: Refer to IBM documentation on the STORAGE macro for a description of the error. Take the appropriate corrective action before continuing with DB2 Manager processing.

02xx An invalid row edit request code has been received from DB2. 'xx' gives the invalid request code.

Action: This indicates either an internal DB2 Manager error or a possible DB2 error. Report the problem to your local DB2 Manager support representative

03xx Error received from TCBTOKEN macro request. 'xx' gives the hexadecimal return code received from the macro call.

Action: Refer to IBM documentation on the TCBTOKEN macro for a description of the error. Take the appropriate corrective action before continuing with DB2 Manager processing.

0500 There is no free request element available for processing the request.

Action: DB2 Manager allows up to 1024 migrated row access requests to be processed concurrently. If this limit is exceeded, the error documented here will be received. In this case, the request should be retried at a less busy time. If the number of concurrent requests in the system is less than this figure, report the problem to your local DB2 Manager support representative.

06xx Error received from ALESERV macro request. 'xx' gives the hexadecimal return code received from the macro call.

Action: Refer to IBM documentation on the ALESERV macro for a description of the error. Take the appropriate corrective action before continuing with DB2 Manager processing.

09xx Error received from LOCASCB macro request. 'xx' gives the hexadecimal return code received from the macro call.

Reason code (hex) ***Explanation***

Action: Refer to IBM documentation on the LOCASCB macro for a description of the error. Take the appropriate corrective action before continuing with DB2 Manager processing.

0A00 A request has been unable to obtain exclusive access to the DB2 Manager control region scheduler lock within the time interval set in the READTIMEOUT or WRITETIMEOUT initialization parameter (depending on type of request).

Action: This condition normally indicates a problem with the control region. Issue a series of operator summary display commands (see page 51) to display the current condition of the scheduler lock. If this indicates that it is unlocked, re-try the request. If it is consistently locked, review the control region joblog for other error messages. If no reason for the error can be identified, report the problem to your local DB2 Manager support representative. To relieve the condition, perform an <u>immediate</u> stop on the control region and restart the region.

0B00 A migrated row has been accessed by the row migration utility OTDBP100.

Action: This is a normal condition when running an DB2 Manager utility program, and will be handled by the program. This error should not occur when executing any other type of program.

0C00 Field TCBUSER was not correctly initialized during DB2 Manager utility processing. The utility job will terminate with an error.

Action: This indicates an internal DB2 Manager error. Report the problem to your local DB2 Manager support representative.

0Dxx A problem has occurred attempting to communicate with the DB2 Manager control region. The second byte of the reason code (‘xx’) will contain a two-digit explanation code, as follows:

01 - An invalid DB2 Manager communications table was found by the row edit procedure OTDBP300.

Action: This indicates an internal error in DB2 Manager processing. Stop and restart the control region to try to clear the error condition. If this does not work, report the problem to your local DB2 Manager support representative.

02 - The name of the DB2 Manager control region held in the ASCB addressed by the row edit procedure OTDBP300 does not match that of the control region at start-up time.

Action: This indicates an internal error in DB2 Manager

Reason code (hex)	***Explanation***
	processing. Stop and restart the control region to try to clear the error condition. If this does not work, report the problem to your local DB2 Manager support representative.
10xx	An error condition has been received by the DB2 Manager control region during processing of the request. The second byte of the reason code contains a two-digit hexadecimal value describing the error. Refer to page 169 for a description of possible control region reason codes.

■ Control region return codes

On completion of a migrated row access request, the DB2 Manager control region issues a two-character return code to the row edit routine OTDBP300, indicating the result of the request. Unsuccessful return codes will result in an SQL –652 error with a reason code of x’10xx’ in field SQLERRD(6) being issued to the SQL caller by DB2 Manager (where ‘xx’ is the error code returned from the control region).

The control region return code is included in the corresponding entry in the request section of the SMF record created during migrated row access processing. The value of the return code may be used by the customer’s own SMF record processing for an analysis of successful and unsuccessful DB2 Manager migrated row access requests.

A return code value of '00' indicates successful storage or retrieval of a migrated row in an Archive Manager database. Non-zero return codes indicate that an abnormal condition was encountered during migrated row access processing. This section lists all possible non-zero values generated by the control region.

Return code (hex value in brackets)	***Explanation***
21 (*15*)	No free tape reader task available to perform migrated row retrieval request, or task processing tape volume containing migrated row is currently at its maximum request level (as set by the MAXQLEN initialization parameter). If a non-zero value of the TAPEWAIT initialization parameter is specified, DB2 Manager will queue the request internally on receipt of this condition, until the request can be processed or until the time interval specified in the parameter has been exceeded, at which point a return code of 21 will be generated.
23 (*17*)	Abend condition intercepted during control region processing.
24 (*18*)	Control region failed to gain exclusive access to a serialized resource within a 1-minute time interval.
28 (*1C*)	Invalid Archive Manager database identifier specified in migrated row access request.
31 (*1F*)	Object containing migrated row not found in Archive Manager database.
64 (*40*)	An error has been received by a control region component when issuing an ALESERV macro, or the name of the DB2 Manager control region held in the ASCB stored with the request does not match that of the control region at start-up time
65 (*41*)	Request has been timed out by control region, because value specified in READTIMEOUT or WRITETIMEOUT initialization parameter has been exceeded.
87 (*57*)	Required Archive Manager database not connected on receipt of migrated row storage request.
88 (*58*)	Archive Manager database connection request received for database which is already connected for update processing. Only one update process may be executed against a single Archive Manager database at any one time.
94 (*5E*)	VSAM error received accessing primary index of Archive Manager database. Message OTD22011 or OTD26010 will be displayed by the control region giving the error and reason codes returned from VSAM.
97 (*61*)	VSAM error received opening primary index of Archive Manager database. Message OTD26010 will be displayed by the control region giving the error and reason codes returned from VSAM.
98 (*62*)	Internal error received during control region processing. Review other error messages issued by the control region for more information about the error condition.
99 (*63*)	Invalid request code received by control region.

Return code (hex value in brackets)	***Explanation***
xx	Any other return code issued by the control region indicates that an error has been received from an Archive Manager call. A message will be issued by the control region giving details of the call and the return and reason codes received from Archive Manager. Refer to the Archive Manager User Manual for a description of this error condition.

Appendices

Appendix A: Sample JCL members

Appendix A gives listings of members on the distributed DB2 Manager sample JCL library.

LCDDB2
Member LCDDB2 is a sample procedure for execution of the DB2 Manager control region started task.

```
//*
//*  Lifecycle Director
//*  DB2 Manager started task procedure.
//*
//*  Change history:
//*  19-Mar-03 GW V1.1 CREATED.
//*  02-Apr-04 GW V1.1 Re-branded.
//*
//*    Edit each of the lines marked "<===" to update:
//*
//*    1.  Name of installed DB2 Manager load library
//*    2.  Name of installed Archive Manager load library
//*    3.  Name of Archive Manager parameter library
//*
//*    If any of the above load libraries are present in the
//*    system linklist, they may be omitted from the STEPLIB
//*    concatenation in this procedure.
//*
//*    Add one OTASxxxx entry for each Archive Manager database
//*    used for storage of migrated rows from DB2, where
//*    "xxxx" is the unique 4-character value used
//*    to identify an Archive Manager database. Each
//*    entry should specify the name of the primary index
//*    dataset for the corresponding Archive Manager database.
//*
//LCDDB2   PROC SIZE=6192K
//*
//LCDDB2   EXEC PGM=OTDBP200,REGION=&SIZE
//STEPLIB  DD  DSN=Database.Manager.load.library,DISP=SHR       <===
//         DD  DSN=Archive.Manager.load.library,DISP=SHR        <===
//SYSPARM  DD  DSN=Database.Manager.parameter.library,DISP=SHR  <===
//SYSPRINT DD SYSOUT=*
//*
//*        Archive Manager database list
//*
//OTASxxxx DD  DSN=Archive.Manager.database.index,DISP=SHR      <===
//*
```

OTDBDEF

Member OTDBDEF creates the DB2 Manager system database.

```
//OTDBDEF  JOB
//*
//*  DB2 Manager system database creation job.
//*
//*  Change history:
//*   GW 19-Mar-03 V1.1 created
//*   GW 02-Apr-04 V1.1 re-branded
//*
//*  This job creates the database manager system database.  It
//*  includes steps to allocate files, define the database and
//*  grant authorities for database access.
//*
//*  Before submitting the job, review each line marked with
//*  with "<===" and modify as appropriate for execution at
//*  your installation.
//*
//STEP01  EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
 DSN  SYSTEM(DB2)                                        <===
 RUN  PROGRAM(DSNTIAD) PLAN(DSNTIAnn) -                  <===
      LIB('DB2.runtime.library')                         <===
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  DROP   DATABASE LCDDB2;
  COMMIT;
//*
//STEP02  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
 DELETE                                               -
 DSN610.DSNDBC.LCDDB2.LCDDB2TS.I0001.A001             -  <===
     CLUSTER
 DELETE                                               -
 DSN610.DSNDBC.LCDDB2.REGIX1.I0001.A001               -  <===
     CLUSTER
 DELETE                                               -
 DSN610.DSNDBC.LCDDB2.REGIX2.I0001.A001               -  <===
     CLUSTER
 SET LASTCC=0
 SET MAXCC=0
 DEFINE CLUSTER                                       -
 (NAME(DSN610.DSNDBC.LCDDB2.LCDDB2TS.I0001.A001)      -  <===
         LINEAR                                       -
         SHAREOPTIONS(3 3)                            -
         VOLUMES(vvvvvv)                              -  <===
         CYLINDERS(1 1)                               -
         UNIQUE   )                                   -
       DATA                                           -
 (NAME(DSN610.DSNDBD.LCDDB2.LCDDB2TS.I0001.A001))        <===
 DEFINE CLUSTER                                       -
 (NAME(DSN610.DSNDBC.LCDDB2.REGIX1.I0001.A001)        -  <===
         LINEAR                                       -
         SHAREOPTIONS(3 3)                            -
         VOLUMES(vvvvvv)                              -  <===
         CYLINDERS(1 1)                                -
         UNIQUE   )                                    -
       DATA                                            -
 (NAME(DSN610.DSNDBD.LCDDB2.REGIX1.I0001.A001))           <===
 DEFINE CLUSTER                                        -
 (NAME(DSN610.DSNDBC.LCDDB2.REGIX2.I0001.A001)         -  <===
         LINEAR                                        -
         SHAREOPTIONS(3 3)                             -
         VOLUMES(vvvvvv)                               -  <===
         CYLINDERS(1 1)                                -
         UNIQUE   )                                    -
       DATA                                            -
 (NAME(DSN610.DSNDBD.LCDDB2.REGIX2.I0001.A001))           <===
//*
//STEP03  EXEC PGM=IKJEFT01,DYNAMNBR=20,COND=(4,LT,STEP02)
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD *
 DSN  SYSTEM(DB2)                                         <===
 RUN  PROGRAM(DSNTIAD) PLAN(DSNTIAnn) -                   <===
      LIB('DB2.runtime.library')                          <===
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  CREATE DATABASE LCDDB2;
  COMMIT;
  CREATE TABLESPACE LCDDB2TS
         IN         LCDDB2
         USING      VCAT DSN610                           <===
         LOCKSIZE   ANY
         CLOSE      NO
         SEGSIZE    64
         BUFFERPOOL BP0;
  COMMIT;
  CREATE TABLE LCDDB2.PLAN_TABLE
               ( QUERYNO          INTEGER      NOT NULL,
               QBLOCKNO           SMALLINT     NOT NULL,
               APPLNAME           CHAR(8)      NOT NULL,
               PROGNAME           CHAR(8)      NOT NULL,
               PLANNO             SMALLINT     NOT NULL,
               METHOD             SMALLINT     NOT NULL,
               CREATOR            CHAR(8)      NOT NULL,
               TNAME              CHAR(18)     NOT NULL,
               TABNO              SMALLINT     NOT NULL,
               ACCESSTYPE         CHAR(2)      NOT NULL,
               MATCHCOLS          SMALLINT     NOT NULL,
               ACCESSCREATOR      CHAR(8)      NOT NULL,
               ACCESSNAME         CHAR(18)     NOT NULL,
               INDEXONLY          CHAR(1)      NOT NULL,
               SORTN_UNIQ         CHAR(1)      NOT NULL,
               SORTN_JOIN         CHAR(1)      NOT NULL,
               SORTN_ORDERBY      CHAR(1)      NOT NULL,
               SORTN_GROUPBY      CHAR(1)      NOT NULL,
               SORTC_UNIQ         CHAR(1)      NOT NULL,
               SORTC_JOIN         CHAR(1)      NOT NULL,
               SORTC_ORDERBY      CHAR(1)      NOT NULL,
               SORTC_GROUPBY      CHAR(1)      NOT NULL,
               TSLOCKMODE         CHAR(3)      NOT NULL,
               TIMESTAMP          CHAR(16)     NOT NULL,
               REMARKS            VARCHAR(254) NOT NULL,
               PREFETCH           CHAR(1)      NOT NULL WITH DEFAULT,
               COLUMN_FN_EVAL     CHAR(1)      NOT NULL WITH DEFAULT,
               MIXOPSEQ           SMALLINT     NOT NULL WITH DEFAULT,
               VERSION            VARCHAR(64)  NOT NULL WITH DEFAULT,
               COLLID             CHAR(18)     NOT NULL WITH DEFAULT,
               ACCESS_DEGREE      SMALLINT,
               ACCESS_PGROUP_ID   SMALLINT,
               JOIN_DEGREE        SMALLINT,
               JOIN_PGROUP_ID     SMALLINT,
               SORTC_PGROUP_ID    SMALLINT,
               SORTN_PGROUP_ID    SMALLINT,
               PARALLELISM_MODE   CHAR(1),
               MERGE_JOIN_COLS    SMALLINT,
               CORRELATION_NAME   CHAR(18),
               PAGE_RANGE         CHAR(1)      NOT NULL WITH DEFAULT,
               JOIN_TYPE          CHAR(1)      NOT NULL WITH DEFAULT,
               GROUP_MEMBER       CHAR(8)      NOT NULL WITH DEFAULT,
               IBM_SERVICE_DATA   VARCHAR(254) NOT NULL WITH DEFAULT,
               WHEN_OPTIMIZE      CHAR(1)      NOT NULL WITH DEFAULT,
               QBLOCK_TYPE        CHAR(6)      NOT NULL WITH DEFAULT,
               BIND_TIME          TIMESTAMP    NOT NULL WITH DEFAULT,
               OPTHINT            CHAR(8)      NOT NULL WITH DEFAULT,
               HINT_USED          CHAR(8)      NOT NULL WITH DEFAULT,
               PRIMARY_ACCESSTYPE CHAR(1)      NOT NULL WITH DEFAULT
               )
               IN LCDDB2.LCDDB2TS;
  CREATE TABLE LCDDB2.REGISTRY
               ( TABNAME          VARCHAR(27)  NOT NULL,
               LCDDBID            CHAR(4)      NOT NULL
               )
               IN LCDDB2.LCDDB2TS;
  CREATE UNIQUE INDEX LCDDB2.REGIX1
                 ON LCDDB2.REGISTRY
                 (
                  TABNAME  ASC
                 )
                 USING      VCAT DSN610                  <===
                 CLOSE      NO
                 BUFFERPOOL BP1
                 PCTFREE    10;
  CREATE UNIQUE INDEX LCDDB2.REGIX2
                 ON LCDDB2.REGISTRY
                 (
                  LCDDBID  ASC
                 )
                 USING      VCAT DSN610                  <===
                 CLOSE      NO
                 BUFFERPOOL BP1
                 PCTFREE    10;
  COMMIT;
  GRANT CREATETAB
        ON DATABASE LCDDB2
        TO PUBLIC;
  GRANT USE OF TABLESPACE LCDDB2.LCDDB2TS
        TO PUBLIC;
  GRANT SELECT, DELETE, INSERT, UPDATE
        ON TABLE LCDDB2.PLAN_TABLE
        TO PUBLIC;
  COMMIT;
```

OTDBPKG

Member OTDBPKG creates the DB2 Manager DB2 application packages.

```
//OTDBPKG  JOB
//*
//*  DB2 Manager DB2 package creation job.
//*
//*  Change history:
//*   GW 19-Mar-03 V1.1 created
//*   GW 02-Apr-04 V1.1 re-branded
//*   GW 30-Apr-04 V1.1 OTDBL130 added
//*
//*  This job creates the DB2 application packages
//*  required for DB2 Manager implementation.
//*
//*  Amend the job as follows before submitting:
//*
//*  1.  Alter the STEPLIB DD card to specify name of the
//*      DB2 runtime load library on the host system.
//*
//*  2.  Change the name of the DB2 subsystem (DB2) in
//*      the DSN statement to match the subsystem name
//*      in use on the host system.
//*
//*  3.  Change the dataset name on the DBRMLIB card
//*      to specify the name of the DB2 Manager DBRM
//*      library unloaded from file 2 of the distribution tape.
//*
//BIND     EXEC PGM=IKJEFT01,DYNAMNBR=20
//STEPLIB  DD   DSN=DB2.runtime.library,DISP=SHR
//DBRMLIB  DD   DSN=Database.Manager.DBRM.library,DISP=SHR
//SYSTSPRT DD   SYSOUT=*
//SYSPRINT DD   SYSOUT=*
//SYSUDUMP DD   SYSOUT=*
//SYSTSIN  DD   *

   DSN SYSTEM(DB2)

   BIND PACKAGE(OTDBL100)                -
        MEMBER(OTDBL100)                 -
        ACTION(REPLACE)                  -
        VALIDATE(BIND)                   -
        ISOLATION(CS)                    -
        FLAG(E)                          -
        SQLERROR(CONTINUE)               -
        RELEASE(COMMIT)                  -
        EXPLAIN(NO)                      -
        KEEPDYNAMIC(YES)

   BIND PACKAGE(OTDBL120)                -
        MEMBER(OTDBL120)                 -
        ACTION(REPLACE)                  -
        VALIDATE(BIND)                   -
        ISOLATION(CS)                    -
        FLAG(E)                          -
        SQLERROR(CONTINUE)               -
        RELEASE(COMMIT)                  -
        EXPLAIN(NO)                      -
        KEEPDYNAMIC(YES)

   BIND PACKAGE(OTDBL130)                -
        MEMBER(OTDBL130)                 -
        ACTION(REPLACE)                  -
        VALIDATE(BIND)                   -
        ISOLATION(CS)                    -
        FLAG(E)                          -
        SQLERROR(CONTINUE)               -
        RELEASE(COMMIT)                  -
        EXPLAIN(NO)                      -
        KEEPDYNAMIC(YES)

   BIND PACKAGE(OTDBL140)                -
        MEMBER(OTDBL140)                 -
        ACTION(REPLACE)                  -
        VALIDATE(BIND)                   -
        ISOLATION(CS)                    -
        FLAG(E)                          -
        SQLERROR(CONTINUE)               -
        RELEASE(COMMIT)                  -
        EXPLAIN(NO)                      -
        KEEPDYNAMIC(YES)

   BIND PACKAGE(OTDBL170)                -
        MEMBER(OTDBL170)                 -
        ACTION(REPLACE)                  -
        VALIDATE(BIND)                   -
        ISOLATION(CS)                    -
        FLAG(E)                          -
        SQLERROR(CONTINUE)               -
        RELEASE(COMMIT)                  -
        EXPLAIN(NO)                      -
        KEEPDYNAMIC(YES)
/*
```

OTDBPLAN

Member OTDBPLAN creates the DB2 application plan required for execution of DB2 Manager.

```
//OTDBPLAN JOB
//*
//*  DB2 Manager DB2 plan creation job.
//*
//*  Change history:
//*   GW 19-Mar-03 V1.1 created
//*   GW 02-Apr-04 V1.1 re-branded
//*   GW 30-Apr-04 V1.1 OTDBL130 added
//*
//*  This job creates the DB2 application plan
//*  required for DB2 Manager implementation.
//*
//*  Review lines marked with a "<===" before submitting
//*  and amend as follows:
//*
//*  1.  Alter the STEPLIB DD cards to specify name of the
//*      DB2 runtime load library on the host system.
```

```
//*
//*  2.  Change the name of the DB2 subsystem (DB2) in
//*      the DSN statements to match the subsystem name
//*      in use on the host system.
//*
//*  3.  Replace the character string 'pppppppp' with
//*      the name of the DB2 plan to be generated.
//*
//*  4.  Change the plan name in the RUN statement (DSNTIAnn)
//*      to match the version of DB2 in use on the host
//*      system.
//*
//BIND     EXEC PGM=IKJEFT01,DYNAMNBR=20
//STEPLIB  DD   DSN=DB2.runtime.library,DISP=SHR        <===
//SYSTSPRT DD   SYSOUT=*
//SYSPRINT DD   SYSOUT=*
//SYSTSIN  DD   *

   DSN SYSTEM(DB2)                                      <===

   BIND PLAN(pppppppp)                   -              <===
        PKLIST(OTDBL100.OTDBL100,        -
               OTDBL120.OTDBL120,        -
               OTDBL130.OTDBL130,        -
               OTDBL140.OTDBL140,        -
               OTDBL170.OTDBL170)        -
        ACTION(REPLACE)                  -
        RETAIN                           -
        VALIDATE(BIND)                   -
        ISOLATION(CS)                    -
        FLAG(E)                          -
        ACQUIRE(USE)                     -
        RELEASE(COMMIT)                  -
        KEEPDYNAMIC(YES)                 -
        EXPLAIN(NO)
/*
//*
//GRNT     EXEC PGM=IKJEFT01,DYNAMNBR=20,COND=(4,LT)
//STEPLIB  DD   DSN=DB2.runtime.library,DISP=SHR        <===
//SYSTSPRT DD   SYSOUT=*
//SYSPRINT DD   SYSOUT=*
//SYSTSIN  DD   *
   DSN SYSTEM(DB2)                                      <===
   RUN PROGRAM(DSNTIAD) PLAN(DSNTIAnn)                  <===
//SYSIN    DD   *
   GRANT EXECUTE
         ON PLAN pppppppp                               <===
         TO PUBLIC;
   COMMIT;
/*
```

Index

Reader’s Comment Form

■ Contact Us

Submit your questions, comments, and suggestions to StorageTek’s Global Learning Solutions. We appreciate your correspondence and are committed to responding to you.

Publication Information

Publication Name:

Publication Part Number:

Questions and Comments:

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form. If you would like a reply, please supply your name and address on the reverse side of this form.

Thank you for your cooperation. No postage stamp is required if mailed in the U.S.A.

TO COMPLY WITH POSTAL REGULATIONS, FOLD EXACTLY ON DOTTED LINES AND TAPE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 2 LOUISVILLE, CO U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

GLOBAL LEARNING SOLUTIONS MS 3256
STORAGE TECHNOLOGY CORPORATION
ONE STORAGETEK DRIVE
LOUISVILLE CO 80028-9989
USA

FOLD HERE AND TAPE DO NOT STAPLE FOLD HERE AND TAPE

If you would like a reply, please print:

Your Name:____
Company Name:___________________________Department:________________________
Street Address:____
City:____
State____ Zip
Code:_________________________

Storage Technology Corporation
One StorageTek Drive
Louisville, CO 80028-3256
USA

NEED MORE INFORMATION?

www.storagetek.com

ABOUT STORAGETEK

Storage Technology Corporation (NYSE: STK) is a $2 billion global company that enables businesses, through its information lifecycle management strategy, to align the cost of storage with the value of information. The company's innovative storage solutions manage the complexity and growth of information, lower costs, improve efficiency and protect investments. For more information, visit www.storagetek.com, or call 1.800.275.4785 or 01.303.673.2800.

WORLD HEADQUARTERS

Storage Technology Corporation
One StorageTek Drive
Louisville, Colorado 80028 USA
1.800.678.4430 or 01.303.673.4430

StorageTek equipment is manufactured from new parts, or new and used parts. In some cases, StorageTek equipment may not be new and may have been previously installed. Regardless, StorageTek's standard warranty terms apply, unless the equipment is specifically identified by StorageTek as "used" or "refurbished."

Replacement parts provided under warranty or any service offering may be either new or equivalent-to-new, at StorageTek's option. Specifications/features may change without notice.