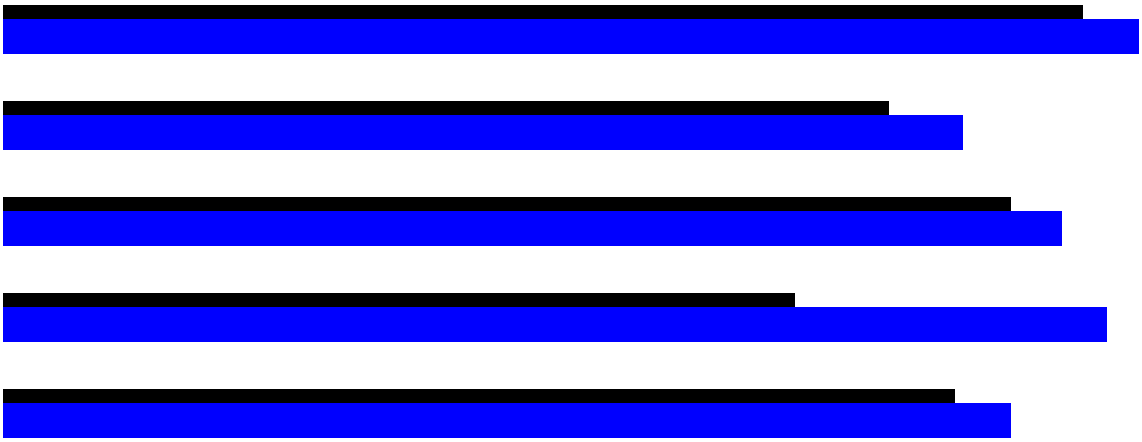
Model 204®



Release Notes

Version 7 Release 1.0

RLN71B

Computer Corporation of America

Corporate Headquarters:		*CCA International:*	
200 West Street 3rd Floor West Waltham, MA 02451 USA		First Floor, Edinburgh House 43-51 Windsor Road Slough, Berkshire SL1 2EE England	
Phone:	(781) 466-6601	Phone:	+44-1753-472800
Fax:	(781) 466-6641	Fax:	+44-1753-472888

Model 204 Release Notes Version 7 Release 1.0
Document Number RLN71B
May 13, 2010

Contents

Model 204®
Release Notes

A Statistics in 64-bit Architecture

Index

Preface

This document contains information about the contents of V7R1.0 of Model 204 and how to use it.

Audience

The audience for this document includes Model 204 end-users, application developers, installers, system managers, and file managers.

Introducing Model 204 electronic documentation

Model 204 documentation includes other manuals to which you might want to refer. Each CD-ROM, titled *Model 204 Documentation,* contains the complete commercially released documentation for a particular release of Model 204.

The document files are in portable document format (PDF) and compiled into a library so you can search all the documents at once for a word or phrase. You can download the manuals to a network server and view them, or view them directly from the CD-ROM. You can view and search an individual manual with any PDF reader, but to search the compiled library, you must download the free Adobe® Acrobat® Reader® from the Adobe web site.

Note: You may access the documentation online or print out copies, as needed. However, consistent with the terms of your license agreement, you may not copy or distribute the CD-ROM, or distribute hard-copies to third parties.

Contacting CCA Customer Support

If you need assistance with this product beyond the provided online help and documentation, and you have licensed this product directly from CCA, either call CCA Customer Support at 1-800-755-4222, or access the Customer Support section of the CCA Web site. The Web address is:

`http://www.cca-int.com`

If you have not licensed this product directly from CCA, please consult your vendor.

Notation conventions

This manual uses the following standard notation conventions in statement syntax and examples:

Convention	Description
`TABLE`	Uppercase represents an option that you must enter exactly as shown.
`TABLE tablename`	In text, italics are used for variables and for emphasis. In examples, italics denote a variable value that you must supply. In this example, you must supply a value for *tablename*.
`READ [SCREEN]`	Square brackets (`[]`) enclose an optional argument or portion of an argument. In this case, specify READ or READ SCREEN.
`UNIQUE | PRIMARY KEY`	A vertical bar (`|`) separates alternative options. In this example, specify either UNIQUE or PRIMARY KEY.
`TRUST | NOTRUST`	Underlining indicates the default. In this example, NOTRUST is the default.
`IS {NOT | LIKE}`	Braces (`{}`) indicate that one of the enclosed alternatives is required. In this example, you must specify either IS NOT or IS LIKE.
`item ...`	An ellipsis (`. . .`) indicates that you can repeat the preceding item.
`item ,...`	An ellipsis preceded by a comma indicates that a comma is required to separate repeated items.
`All other symbols`	In syntax, all other symbols (such as parentheses) are literal syntactic elements and must appear as shown.
`nested-key ::= column_name`	A double colon followed by an equal sign indicates an equivalence. In this case, *nested-key* is equivalent to *column_name*.
`Enter your account: sales11`	In examples that include both system-supplied and user-entered text, or system prompts and user commands, boldface indicates what you enter. In this example, the system prompts for an account and the user enters **sales11**.
File > Save As	A right angle bracket (>) identifies the sequence of actions that you perform to select a command from a pulldown menu. In this example, select the Save As command from the File menu.

1

Features and Enhancements

In this chapter

- Overview of new features and enhancements
- Critical migration issues
- This shipment contains
- Installation considerations
- V7R1.0 at a glance
- Performance improvements
- 64-bit architecture for above the bar storage
- User Language $function enhancements
- MSGCTL command enhancement
- SWITCH STREAM command
- Implementing the date/time stamp feature
- Installing the date/time stamp feature
- SUBSYSMGMT enhancements
- Sending e-mail messages via User Language
- Additional features
- Upward compatibility issues

Overview of new features and enhancements

This chapter contains installation and migration information pertinent to Model 204 V7R1.0. Please read through the material for the latest information on contents and installation before you start the actual installation.

Critical migration issues

Migrating directly from V5R1.0 to V7R1.0

If you are migrating from Model 204 V5R1.0 directly to V7R1.0, be sure to study the *Model 204 V6R1.0 Release Notes*, included on the Model 204 V7R1.0 Documentation Library CD, as well as available for direct download from the CCA web site.

All the information in the *Model 204 V6R1.0 Release Notes* has been incorporated into the documentation set and therefore, scattered through out. You must make yourself aware of the functionality and features that were introduced in V6R1.0 to understand how both V6R1.0 and V7R1.0 will affect your development decisions, performance, and daily operations.

This shipment contains

- CPU ID zap
- Decryption keys
- *Model 204 Release Notes V7R1.0*
- An installation guide for your operating system
- Tape map
- Mainframe cartridge for your operating system:

Operating System	Contents	Label
z/OS	Nucleus and install	`MV7100`
z/VM	Nucleus and install	

Using the CCA Web site

Early Warnings and object replacement modules are available via the CCA web site. On the CCA web site either you can request an Autofix Tape containing them or you can directly download Autofix, individual Early Warnings, and object replacement modules.

Accessing the CCA web site

The following steps access the Technical Support maintained on the CCA Web site.

1. Go to:

   `http://www.cca-int.com`

2. Select the Customer Support button on the menu bar.
3. Select the Technical Support link.

Requesting an Autofix tape

Although Autofix, Early Warnings, and object replacement modules are posted on the CCA web site, you may request an Autofix tape taking the following steps:

1. Select the Ordering Model 204 Maintenance link on the Technical Support page.
2. Click Request Autofix Tape.
3. Complete the Autofix Order Form and click OK.

Downloading directly from the Web site

You can enter the secured support area on the Technical Support page to access current maintenance and other support activities. You need a User ID and password from CCA to access this area. If you do not have your access information, please send an e-mail request to:

`maintinfo@cca-int.com`

Once you have gained access to the secured support area on the Model 204 Customer Support Services page, you may download the latest Autofix release or access Early Warnings and object replacement modules. Select Autofix, Early Warnings & Late Breaking News and follow the navigation provided.

For example, in the Autofix, Early Warnings & Late Breaking News link select the Early Warnings link. Then select the Model 204 Version 7 Release 1.0 link.

Installation considerations

Using INS204 updates for this release

As always you must use only the installation jobs on this tape for this release. In particular, changes have been made to the installation process for z/OS that you must use the distributed INSPARMS and M204DECR jobs for this release installation. You must not use installation jobs from an earlier release.

Support for CMS formatted Model 204 databases dropped in V7R1.0

Because of the 64-bit changes introduced into Model 204, CMS formatted databases can no longer be supported. Attempts to bring up an Online using these types of databases results in *Unsupported Device Type* messages. CCA recommends that you reorganize such files to an OS formatted disk.

Operating system requirements

Model 204 V7R1.0 requires the following operating system support:

- For z/OS, z/OS Version 1 Release 02.0 or later.
- For z/VM, z/VM Version 5 Release 4.0 or later.

Hardware requirements

Model 204 V7R1.0 requires IBM zSeries processors.

Changes to INSPARMS in V7R1.0

With Model 204 V7R1.0, a number of obsolete options were removed from INSPARMS to simplify the installation process. Removed are references to options, features, and jobs which relate to the Advantage/SQL features and ACCESS/204 and TARGET/204 products. As always, the INSPARMS option member delivered with this release of Model 204 should be the only one used when installing this release of Model 204.

Changes to user-written $functions and code tables

Your installation might have locally written and maintained assembler code that is linked with the Model 204 nucleus. The supported extensions are:

Supported extensions	Description
CDTB	Code tables
CUST	Code you can customize (Assembler source like FUNU)
FLODXTn	FLOD exits

Supported extensions	Description
FUNU	Local $functions
M204CKPX	Checkpoint exit. Exit M204CKPX can be entered only on completion of successful transaction checkpoints. If an extended quiesce has been specified, the exit will be taken just before the extended quiesce.
M204JLOG	SWITCH command user exit. Exit M204JLOG gains control when a SWITCH command is issued against a CCAJLOG GDG stream.
MSGU	Local messages
X3270CHK	Check for terminal characteristics
X3270IN	Convert input to 3270 format
X3270OUT	Convert output from 3270 format

Whenever you upgrade Model 204 to a new release, you must reassemble all your local extensions using the latest Model 204 macro library that is copied from the install package for linking with the new release.

SVC compatibility

If you use SVCs, you must install new SVCs for each new release of Model 204. CCA does not support transferring SVCs from a previous release into a newer release. Be sure to update M204XSVC and M204XDM.

Supporting TPROCESS COBOL applications

In the CICSLINK job you must relink any CICS TPROCESS applications using AMODE=31, as AMODE=24 is no longer supported.

Reassemble MSGU

The new MSGCTL implementation requires a reassembly of the user message module (MSGU). This may also include third-party software messages that you install.

IFAM1, IFAM2, and IFAM4 jobs: Compiling under Enterprise PL/I for z/OS

When compiling a PL/I application under the Enterprise PL/I for z/OS compiler, the following compiler parameter is required:

```
DEFAULT(LINKAGE(SYSTEM))
```

This causes the parameter list to be built in the same way that it was built by the old compilers, including turning on the high-order bit of the address of the last parameter.

For example:

```
//PLICMPL EXEC PGM=IBMZPLI,PARM='OBJECT,OPTIONS,
//             DEFAULT(LINKAGE(SYSTEM))',REGION=512K,...
```

If this compiler option is not specified, subsequent executions of the application will fail with 0C4 abends.

MERGEJ module

The MERGEJ module is automatically linked AMODE=31. You do not need to make any changes to your installation process.

TSFPROC: File organization changed

TSFPROC, the procedure file for the CCATSF application subsystem, has occasionally filled due to its use as both the procedure file and as a temporary work file. To avoid this problem, TSFPROC has been redefined with FILEORG bit X'24', meaning Unordered, Reuse Record Number, and BSIZE has been increased.

CCA recommends that you restore the new version of TSFPROC from the V7R1.0 distribution tape at installation, although you may do this update at a later time.

For a z/OS operating system

During an install or reinstall process, ensure that the SQLINST job will be generated by specifying INSTALL-SQL=I or R in the INSPARMS member of the INS204.JCLLIB partitioned dataset. This action will generate jobs needed for a new allocation, creation, and population of the improved TSFPROC file.

For the z/VM operating system

Edit and run the SQLINST job.

M204CKPX checkpoint user exit

The checkpoint user exit, M204CKPX, can be invoked if linked in. This exit runs after the CHECKPOINT records are written, and just before the CHECKPOINT COMPLETED message is issued. All Model 204 databases are physically consistent on disk at this time, because all updated pages have been flushed to disk to prepare for checkpoint processing.

The exit runs *before* any update users are allowed to run in the Online, which allows users to write an exit that backs up all their database files, between the hours of *x* and *y*, all based on the user exit code.

When the exit abends or completes, the Online releases updating users to run and continues. Because this exit is invoked each time a checkpoint is taken, you must forego any extensive processing until the Online is in a low use period, then back up the files.

Example The following example is a M204CKPX ASSEMBLER exit. You can use the shell of the following program to write your own user exit. If your user exit abends, Model 204 tries to continue. All registers can be used. You can safely copy the files because the modified pages have all been flushed to disk and are physically consistent. The Online continues to service read-only users, and updating is suspended until this exit completes. You might want to include your own ESTAE exit macro to deal with abends.

A sample M204CKPX ASSEMBLER exit:

```
M204CKPX  CSECT
M204CKPX  AMODE 31
M204CKPX  TITLE 'TEST THE MODEL 204 CHECKPOINT USER EXIT'
X10       EQU    10
X11       EQU    11
X12       EQU    12
X13       EQU    13
X14       EQU    14
X15       EQU    15
          STM   X14,X12,12(X13)   SAVE CALLERS REGISTERS
          LR    X12,X15           ESTABLISH BASE REGISTER
         USING  M204CKPX,X12
          LA    X10,SAVEAREA      GET A(LOCAL REGISTER SAVEAREA)
          ST    X10,8(,X13)       CHAIN OUR SAVEAREA TO CALLERS
          ST    X13,SAVEAREA+4    CHAIN CALLERS SAVEAREA TO OURS
          LA    X13,SAVEAREA      SET A(OUR SAVEAREA)
          WTO   'M204CKPX CHECKPOINT EXIT INVOKED,
                UPDATERS SUSPENDED'

* ******************************************************** *
* * Add code here that calls routines to back up your      * *
* * Model 204 databases.                                   * *
* *                                                        * *
*   DC    X'000000000000'    TEST TO SEE WHAT HAPPENS WHEN * *
*                            THE USER EXIT ABENDS          * *
* ******************************************************** *
          WTO   'M204CKPX EXIT ENDING, UPDATERS WILL BE RELEASED'
           L     X13,4(,X13)        RESTORE CALLERS SAVE AREA ADDRESS
           ST    X10,16(X13)        SET RETURN CODE (R15)
           LM    X14,X12,12(X13)    RESTORE CALLERS REGISTERS
           BR    X14                RETURN TO CALLER
           DS    0D
SAVE AREA  DS    18F                REGISTER SAVE AREA
          LTORG
           END
```

If Model 204 detects an abend in the M204CKPX exit, the following message is written to the JES log or is displayed on the z/VM console:

```
M204.CKPX: ABEND IN M204CKPX IGNORED, ATTEMPTING TO CON-
TINUE
```

Note: If M204CKPX abends, Model 204 attempts to continue processing, but there is no guarantee that it can do so. Further processing might be prevented for one of the following reasons.

- M204CKPX destroyed storage that Model 204 depends on.
- Left interrupts disabled, or did not restore the ESTAE or ESPIE macro routines.

M204JLOG assembler exit

The assembler exit, M204JLOG, can be invoked if linked in. This exit is initiated when a SWITCH command is issued against a CCAJLOG GDG stream. The exit may be any AMODE and need not be reentrant.

Coding considerations

On entry, the registers contain:

R4 = A(GDG LIOD)

R9 = A(new switch control record) (see CRD dsect)

R13 = A(OSW save area)

R14 = return address

R15 = base address

Model 204 expects no output from the exit. All registers must be restored before return. If any Model 204 data structures are modified by the exit, unpredictable results may occur.

Note: If the user exit abends, Model 204 issues an error and produces a snap. This leaves Model 204 waiting for the switch to complete. At that point the switch will never complete, no further Online activity is possible, and the Online must be cancelled.

In addition, while the M204JLOG user exit is running, Model 204 cannot continue normal processing until a return from the exit is accomplished. For this reason, WAITS and I/Os inside the exit are strongly discouraged.

Sample M204JLOG Assembler exit

```
M204JLOG  CSECT
M204JLOG  AMODE 31
M204JLOG  TITLE 'TEST THE MODEL 204 CCAJLOG USER EXIT'
X10       EQU   10
X11       EQU   11
X12       EQU   12
X13       EQU   13
X14       EQU   14
X15       EQU   15
```

```
         STM   X14,X12,12(X13)    SAVE CALLERS REGISTERS
         LR    X12,X15            ESTABLISH BASE REGISTER
        USING  M204JLOG,X12
         LA    X10,SAVEAREA       GET A(LOCAL REGISTER SAVEAREA)
         ST    X10,8(,X13)        CHAIN OUR SAVEAREA TO CALLERS
         ST    X13,SAVEAREA+4     CHAIN CALLERS SAVEAREA TO OURS
         LA    X13,SAVEAREA       SET A(OUR SAVEAREA)
         WTO   'M204JLOG EXIT INVOKED, DOING SOMETHING'
* ************************************************************ *
* * CUSTOMERS MAY PLACE CODE HERE TO DO WHATEVER THEY DESIRE.   * *
* ************************************************************ *
         WTO   'M204JLOG EXIT ENDING'
         L     X13,4(,X13)        RESTORE CALLERS SAVEAREA ADDRESS
         ST    X10,16(X13)        SET RETURN CODE (R15)
         LM    X14,X12,12(X13)    RESTORE CALLERS REGISTERS
         BR    X14                RETURN TO CALLER
         DS    0D
SAVEAREA DS    18F                REGISTER SAVE AREA
        LTORG
         END
```

Model 204 compatibility with operating systems

Check the CCA web site for the most current listing of IBM mainframe computers that are compatible with Model 204 V7R1.0:

`www.cca-int.com`

Beginning on the home page select:

`Customer Support > Technical Support > Model 204 Operating System Support`

V7R1.0 at a glance

Table 1-1 highlights the new features, improved performance, and new functionality in Model 204 V7R1.0

Table 1-1. New features and functionally in V7R1.0

New feature	Briefly…	Op. sys.	For details see…
64-Bit support	Support for above the bar data structures, including related parameters and statistics	z/OS z/VM	“64-bit architecture for above the bar storage” on page 1-18
Maximizing base record storage to 16 million.	Table B for base records and Table X for extension records.	z/OS z/VM	“Storing base records and extension records in separate tables” on page 2-3
COMPACTB enhancements	Improved compaction and performance for COMPACTB	z/OS z/VM	“COMPACTB data compaction enhancements” on page 2-10
Tables B, D, and X can be increased automatically	Using parameters BAUTOINC, DAUTOINC, and XAUTOINC to enable automatic increases.	z/OS z/VM	“Automatic increase for Tables B, D, and X” on page 2-13
Date/time stamp feature	Tracking date and time of record update on updated records.	z/OS z/VM	“Implementing the date/time stamp feature” on page 1-32 “Installing the date/time stamp feature” on page 1-34 Chapter 3: Date/Time Stamp Support
Send e-mail messages with a single function call	$SNDMAIL function formats and addresses e-mail messages with ease using specified arguments	z/OS z/VM	“Sending e-mail messages via User Language” on page 1-40
Improved field extraction performance	Set FILEORG to X’80’ to improve field extraction performance	z/OS z/VM	“FILEORG bit X’80’ to improve nonpreallocated field extraction” on page 1-48
Performance improvements	Reduce CPU usage Reduce disk page requests Reducing number of disk buffer I/O control blocks	z/OS z/VM	“CPU reduction” on page 1-16 “Reduced requests for page reads (DKPR statistic)” on page 1-16 “MAXSIMIO to save IOS Branch Entry storage” on page 1-48
Large dataset support	A dataset for a database file or journal may have more than64K tracks	z/OS z/VM	“Large dataset support” on page 1-52 and “DEFINE DATASET: Large dataset characteristics” on page 6-36
COMPACTE, Table E data compactor	Reduce Table E record fragmentation	z/OS z/VM	“COMPACTE data compaction for Table E” on page 2-17

Table 1-1. New features and functionally in V7R1.0 (continued)

New feature	Briefly…	Op. sys.	For details see…
Regenerate with incomplete set of journals	REGENERATE IGNORE option eliminates full journal set requirement	z/OS z/VM	“Running REGENERATE with the IGNORE argument” on page 1-54
MERGEJ processing enhancements	MERGEJ recognizes journal type for improved performance	z/OS z/VM	“MERGEJ support for journal types” on page 1-59
SWITCH STREAM command	Force switch to next member of CHKPOINT, CHKPNTS, CCAJRNL, or CCAJLOG stream	z/OS z/VM	“SWITCH STREAM command” on page 1-28 “SWITCH STREAM: Change stream to next member” on page 6-60
STREAM configuration support for CHKPOINT and CHKPNTS datasets	Defining checkpoint datasets as streams	z/OS z/VM	“Checkpoint configuration support” on page 1-53 “CHECKPOINT: Requesting and handling a checkpoint” on page 6-29
MONITOR command enhancements	MONITOR command reports disk buffer size, including Table X. You can use an M to issue a MONITOR command. MONITOR SIZE provides definition and usage for journal and checkpoint streams.	z/OS z/VM	“Enhanced MONITOR command for disk buffers” on page 1-22 “MONITOR command synonym” on page 1-53 “MONITOR SIZE: Tracking journal and checkpoint streams” on page 6-47
Each disk buffer has increased by a eight bytes to enhance detection and better handle a write to a buffer that overruns the disk buffer.	Each buffer has extra bytes for flagging overruns.	z/OS z/VM	“Disk buffer overrun detection” on page 1-50
Improved MSGCTL command for handling error messages	Manage message processing based on message type and designating messages as counting errors.	z/OS z/VM	“MSGCTL command enhancement” on page 1-26, “MSGCTL: Setting message output” on page 6-48
Customized User Language compilation error handling	Controlling the volume of error messages and their display	z/OS z/VM	“Compilation error message management” on page 1-52 “COMPERR: Compiler errors reporting options” on page 6-5

Table 1-1. New features and functionally in V7R1.0 (continued)

New feature	Briefly…	Op. sys.	For details see…
ERMX processing changes	Processing changes when ERMX exceeded	z/OS z/VM	“ERMX parameter” on page 1-63 “ERMX: Maximum number of errors for user’s session” on page 6-9
CUSTOM=(9) for PRIORITY command	Suppress all output from a PRIORITY command.	z/OS z/VM	“New CUSTOM setting” on page 1-59
ENQCTL enhancement	List all shared DASD enqueues present on the file’s FPL	z/OS z/VM	“ENQCTL command enhanced” on page 1-60
Improved snap options	Date/time stamp added to DUMP symptom record to maintain multiple versions on the system. New parameters SNAPFAIL and SNAPFLIM better manage snap failure events.	z/OS z/VM	“DUMP and SNAP handling” on page 1-16 “SNAPFAIL: Number of CCASNAP failures” on page 6-21 “SNAPFLIM: Snap failure limit” on page 6-21
Storage options for ASPY subsystem saved precompiled procedures	APSYPAGE, TEMPPAGE, and CCATEMP storage for APSY subsystem saved precompiled procedures	z/OS	“DSPOPT: Data- and hiperspace options” on page 6-7
New ERRMSGL parameter	Establish the actual number of bytes used to save error messages.	z/OS z/VM	“ERRMSGL: Number of bytes used for saved error messages” on page 6-10
IP address information for TN3270 Telnet connection	You can display the underlying connection between the PC and Online.	z/OS	“IPADDR: Telnet server IP address display” on page 6-13
FOR RECORD NUMBER/COMMIT record locking changes	COMMIT statements inside of FOR RECORD NUMBER loops release only the lock pending update, single record lock—the transaction back out (TBO) lock.	z/OS z/VM	“COMMIT in FOR and FOR RECORD NUMBER (FRN) loops changed” on page 1-67
User Language $function enhancements	$RESET (a parameter) and $VIEW (VIEW command corollary)	z/OS z/VM	“User Language $function enhancements” on page 1-24
Support for update transactions referencing both TBO and non-TBO files	Issue a COMMIT between each file request and all will compile.	z/OS z/VM	“Updates to TBO and non-TBO files in the same transaction” on page 1-67
Increased number of available found sets	Changes that allow for a greater number of found sets within a user session	z/OS z/VM	“Increased number of available found sets” on page 1-16

Table 1-1. New features and functionally in V7R1.0 (continued)

New feature	Briefly…	Op. sys.	For details see…
SQL enhancements	All table join expressions are supported. SELECT statement column limit removed.	z/OS z/VM	"SQL outer join features expanded" on page 1-60 and "SQL SELECT extended" on page 1-61
MP scalability improvements	MP/204 option to ensure that long request values are no exceeded	z/OS	"SCHDOPT: Scheduler operation and accounting" on page 6-19
MQ/204 enhancements	Support implemented for grouping messages and JMS (Java Message Service Control option provided for retention or release of MQ subtasks for APSY subsystem processing	z/OS	Chapter 4: MQ/204 Enhancements "MQSUBREL: MQ subtask release control" on page 6-16
Dictionary/204 update	Dictionary/204 support for date/time stamps and Table X.	z/OS z/VM	Chapter 5: Dictionary/204

Performance improvements

Reduced requests for page reads (DKPR statistic)

In some cases, pages may now be kept open longer. This may result in fewer page requests. This improvement may lead to a small reduction in CPU and real disk I/O.

Increased number of available found sets

The total number and size of found sets possible at one time in Model 204 has been increased.

DUMP and SNAP handling

The date and time stamp of a DUMP occurrence has been added to the DUMP symptom record. This makes that record unique for Model 204 dumps differing by more than 0.1 second. This uniqueness between dumps could now result in more dumps being retained on your system.

The parameter SNAPFLIM that can be set and reset by a system manager has been introduced to control the number of times CCASNAP processing can fail before CCASNAP processing is deactivated. See “SNAPFLIM: Snap failure limit” on page 6-21. See also “Improved snap management” on page 1-50.

Another parameter, SNAPFAIL, that is view-only is a counter for the number of times the CCASNAP process has failed. When SNAPFAIL reaches SNAPFLIM, SNAPLIM is set to the SNAPID value, preventing further CCASNAP attempts. If SNAPLIM=0 and an unexpected program exception occurs while processing a CCASNAP a 4095 abend still results. See “SNAPFAIL: Number of CCASNAP failures” on page 6-21.

CCASNAP dataset changes

The CCASNAP file attributes have changed from RECFM=VBA, LRECL=125 to RECFM=VBA, LRECL=137. This change was made to accommodate the snapping of 64-bit address ranges, for the extra nine bytes for the high-half address value and separating underscore.

CPU reduction

Model 204 V7R1.0 includes a set of internal code changes designed to provide a significant amount of CPU relief to your applications.

Some of these changes to reduce CPU are:

- Optimized search for screen and image items
- FIND and FOR loop processing improvements

- Reduction in requests for page reads (DKPRs) as a result of improved scheduler handling of open pages
- Elimination of trailing blanks when Image Items are accessed
- Update times reduced by improving start of transaction processing
- MSGCTL

  The new MSGCTL command argument, NOACTION, avoids almost all error message processing. Error messages now have a 1-byte indicator that informs message processing whether the message needs to be processed and if the message has been changed by a MSGCTL command. If a message does not need processing, processing immediately returns to the main code. By this same mechanism, message control chain processing is eliminated for all but MSGCTL changed messages. Both these changes result in decreased CPU consumption.

Storage improvements for APSY subsystem saved precompilation

If not otherwise set, DSPOPT now defaults to X'00'. When a saved compilation page is moved to the APSYPAGE area, it is no longer also kept in CCATEMP, which eliminates duplicate storage.

This is particularly useful when you keep CCATEMP in memory, using the TEMPPAGE parameter. This reduces in memory CCATEMP storage requirements by approximately (N/1.50), where N is the number of 4K-byte pages of APSYPAGE that are used. In memory CCATEMP size is defined by the TEMPPAGE parameter.

Setting DSPOPT=X'80' and TEMPPAGE greater than zero causes saved compilations to be saved in both CCATEMP and APSYPAGE. Although this is wasteful of storage, it does not cause problems. If you set both parameters, you may see a higher use of CCATEMP in storage.

Setting APSYPAGE greater than zero causes APSY subsystem precompiled pages to be saved in storage. If CCATEMP=0 is also set your CCATEMP is kept on disk and all saved compilations are stored in both locations. In memory APSYPAGE storage can still be reduced if DSPOPT is set to include the X'80' bit. With both those settings, Model 204 employs the least recently used (LRU) algorithm to keep the more heavily used saved compilations in memory while the less recently used saved compilations migrate to disk. See "DSPOPT: Data- and hiperspace options" on page 6-7.

Sub-transaction checkpoint processing improvement

For transaction back out files, sub-transaction checkpoint constraint and transaction back out log pages are now written asynchronously.

64-bit architecture for above the bar storage

Understanding above the bar storage

In 64-bit mode z/OS may create a region that has storage above the 2-gigabyte line called the bar. Address spaces with above the bar storage consist of three areas:

Storage location	Used for…
In 0-2 gigabyte range	Programs and data
Above 2 gigabytes (above the bar)	Data only
Note: in 2-4 gigabyte range	Unavailable for any purpose

An address space is 16 exabytes or 2**64-1.

Set the IBM MEMLIMIT system option

To implement above the bar storage, IBM requires that you set a limit on how much of that virtual storage each address space can use. This limit is called MEMLIMIT. If you do not set MEMLIMIT, the system default is 0, meaning no address space can use above the bar virtual storage. To allocate Model 204 data structures such as buffer pool above the bar, MEMLIMIT should be properly set when running this release.

IBM provides several options to override the system default. Use one of the following ways when you install and run V7R1.0:

- SMF MEMLIMIT parameter
- MEMLIMIT on JOB and EXEC statement, or REGION=0M in JCL
- MEMLIMIT environment through IEFUSI

You should refer to IBM documentation for MEMLIMIT and limiting above the bar storage use in z/Architecture to implement the option that best meets your site requirements.

The use of real storage below the 2-gigabyte address is not controlled by MEMLIMIT. Only the amount of virtual storage is controlled by MEMLIMIT.

Model 204 64-bit architecture support

Model 204 V7R1.0 allows nearly unlimited 64-bit virtual storage for the Model 204 buffer pool. Table B and Table X pages can be placed in a buffer pool above the bar. Pages from other tables reside in the buffer pool below the bar and above the line. A considerable portion of below the bar storage is taken by the operating system and other Model 204 areas.

- In most cases, Table B pages constitute the biggest portion of all pages in the buffer pool. Moving Table B pages to an above the bar buffer pool lets Model 204 place more pages from all other tables in the below the bar buffer pool and thereby reduce I/O and CPU time to read and write pages to and from disk.
- When a buffer is allocated above the bar, the corresponding disk buffer control blocks (one per buffer, 160 bytes each) and hash cells (three per buffer, 16 bytes each) are also allocated above the bar. This means there is no below the bar storage penalty for allocating above the bar buffers.
- Having these two buffer pools instead of one improves Model 204 scalability by reducing MP collisions, when using buffer pool resources.
- Eight bytes have been added to the end of every buffer, above and below the bar, to detect buffer overruns. The new buffer size per page is 6192 bytes (or 6184 plus 8).

The following Model 204 entities are accessed directly in above the bar storage:

- CCATEMP in memory

  When using CCATEMP in memory, CCATEMP pages can be accessed directly without going through the buffer pool. The above the bar CCATEMP pages in TEMPPAGE are accessed directly. If TEMPPAGE is non-zero, then the CCATEMP pages are always allocated above the bar, regardless of the setting of NUMBUFG.
- Record set bitmap pages
- Images and Screens

  Pages used for Model 204 SCREEN and IMAGE items now reside in the buffer pool above the 2-gigabyte bar.
- Index pages for Tables C and D
- APSYPAGE
- Swapped out servers

Managing above the bar storage

When NUMBUFG is set to a nonzero value, an above the bar buffer pool is allocated with NUMBUFG buffers. This is in addition to the below the bar buffer pool which is always allocated with at least the minimum number of buffers, calculated as follows:

```
NLRUQ * ((NSERVS + NSUBTKS) * MAXOBUF + 15)
```

Table B and Table X pages use the above the bar buffer pool. Those pages are not read into the below the bar buffer pool. Consequently, most sites can

reduce the size of the below the bar buffer pool by the high water mark of Table B pages currently resident in that buffer pool.

To quickly implement the above the bar feature, initially set NUMBUFG equal to your MAXBUF setting and leave MAXBUF at its current setting.

The minimum number of above the bar buffers calculated by Model 204 uses the following formula:

```
NLRUQG * ((NSERVS + NSUBTKS) * MAXOBUF + 15)
```

If you set NUMBUFG to a lower value, it is reset to the calculated value.

If NUMBUFG is greater than zero, the buffer pool is allocated above the bar. In addition, control blocks associated with above the bar buffers are also allocated above the bar. NUMBUFG is limited to buffer pools of 4.2 terabytes or fewer.

To use above the bar buffer pool in z/OS, IOS Branch is required. This means XMEMOPT must be set to include X'02'. You can explicitly exclude allocating above the bar buffers by setting NUMBUFG=0.

If NUMBUFG is greater than zero and XMEMOPT does not include X'02', the following message is issued, NUMBUFG is not reset, and the job terminates.

```
M204.2581: XMEMOPT=2 (IOS BRANCH) REQUIRED FOR option
```

If you cannot get the number of buffers you requested, the job fails.

Determining NUMBUFG setting

The number of buffers you want to allocate above the bar and below the bar is dependant on the mix of work that is being done on your system. At this time, buffers for Table B, Table X, and some of their related data structures can go above the bar.

- The LDKBMWNG parameter, which applies to above the bar buffers, corresponds to the LDKBMWND parameter, which applies to below the bar buffers.
- If NLRUQG is set greater than 1, then the value of LDKBMWNG is rounded up to a multiple of NLRUQ. LDKBMWND has a minimum size of one (1).

High values of LDKBMWNG might unnecessarily increase the number of writes done (measured by the DKWR statistic). Low values might cause excessive waiting for buffers (measured by the DKSWAIT statistic). CCA recommends starting values for LDKBMWND and LDKBMWNG at 10% of NUMBUF and NUMBUFG, respectively.

If you do not set LDKBMWNG, it is set to the same value as LDKBMWND.

Refer to the *Model 204 System Manager's Guide* for more information about disk buffer monitor parameters and statistics.

HASHCELL - Allocation of hash cells per buffer pool page

You can control the number of hash cells allocated in the hash table with the new parameter HASHCELL. The hash table is used to locate pages in the buffer pool based on the file and page number. The default, and minimum, is three hash cells per page. In this release you can allocate as many as seven hash cells per page. CCA recommends resetting the default only when running with AMPSUBS>0.

Using a higher value will:

- Reduce the number of hash cell collisions and thus, reduce the CPU consumed to resolve any collisions.

  Note: the DKSRHC statistic is not longer collected.

- Result in the use of more virtual storage for the increased number of hash cells. Each hash cell is 16 bytes and HASHCSLL number of hash cells are allocated per buffer. The default value of HASHCELL is 3.

If NUMBUFG is also set to a value greater than zero to allocate buffers above the bar, the hash cells are also allocated above the bar, saving below the bar storage. The total amount of storage required for hash cells can be calculated using the following formula:

```
16 * HASHCELL * (NUMBUF + NUMBUFG)
```

Handling 64-bit statistics

To support very long running Model 204 regions, CCA has modified the capacity of statistical counters by increasing the size of some statistics and also exploiting 64-bit processing where appropriate. For any in-house or third-party support applications that process statistical counters, you will need to review the statistics generated.

As some of the statistics fields are now double words, you should check the *Appendix A: Statistics in 64-bit Architecture* for the new layout of the System, Final and Partial statistics. Also, additional Disk Buffer Monitor, MP/204, and File statistics have been updated.

Look at your in-house or third-party support applications to see if you need to make changes because of the increased length of some of the statistics. Make any changes necessary to your applications, then reassemble with this new release.

Even if your in-house or third-party support applications do not refer to any of these double word statistics, you must reassemble your applications since all statistics offsets have changed.

New and improved MONITOR statistics

The following statistics are for buffers below the bar—DKRDL, DKSAWBL, DKSAWWL, and DKWRL.

Table 1-2. Statistics new in V7R1.0

Statistic	Counts	Type
DKRDL	The number of physical page reads into the below the bar buffer pool	System
DKSAWBL	Number of anticipatory writes from the bottom of the LRU queue for the below the bar buffer pool. The page in the buffer is then deleted and no longer available without a re-read. Usually a small value or zero; typically only incremented by a few rare events. This is not an anticipatory write.	System
DKSAWWL	Anticipatory writes when a buffer was removed from the window below the bar	System
DKWRL	Number of physical page writes from the below the bar pool	System

Enhanced MONITOR command for disk buffers

You can use the MONITOR DISKBUFF commands to analyze the buffer pool utilizations.

- MONITOR DISKBUFF output shows buffer usage combining above and below the bar buffers.
- MONITOR DISKBUFFG output shows buffer pool usage for only above the bar buffers.
- MONITOR DISKBUFFL output shows buffer pool usage for only below the bar buffer usage.

Use these commands throughout the day across varied types of daily and event processing to evaluate your buffer allocations.

Using MONITOR DISKBUFF commands

You can see the types of pages that are in your buffer pools at any point in time using the MONITOR DISKBUFF command. The output of this command displays the types of pages that are in buffers and how many of each type.

MONITOR DISKBUFF

```
FILENAME  FCT    TBLA    TBLB    TBLC    TBLD    TBLE    TBLX   *TOTAL*
--------  ---  ------  ------  ------  ------  ------  ------   -------
CCATEMP     0      40       0       0       0       0       0        40
PROC1       1       0       0       0      11       0       0        12
FILETBLX    1       1      10       0       2       0       2        16
TESTZ       1       1      11       0       3     185       0       201
*TOTAL*     3      42      21       0      16     185       2       269
```

User Language $function enhancements

$RESET function

The $RESET function lets you change selected parameters. Currently, you can reset only the MQSUBREL parameter.

Syntax to reset a parameter value:

```
$RESET(parameter_name, parameter_value)
```

The return code has the following meanings:

Code	Means: Parameter…
0	Was reset successfully
1	Name is incorrect or missing
2	Value is incorrect or missing
3	Is not in the user section of KOMM
4	Cannot be reset by User Language
5	Cannot be reset by any user
6	Requires special routine to handle

Return codes 1 and 2 indicate a programming error must be corrected.

Return codes 3 to 6 indicate system errors that may need to be reported to CCA Customer Support.

Usage example

```
%A = $RESET('MQSUBREL',1)
IF %A > 0 THEN
  PRINT 'Parameter not reset. Return code =  ' %A
END IF
```

$VIEW arguments for journal and checkpoint

Syntax to view the size and use of journal and checkpoint streams

The format of $VIEW for the use and size of journal and checkpoint streams is:

```
$VIEW('CCAJRNL'|'CCAJLOG'|'CHKPOINT'|'CHKPNTS')
```

Specify the following:

```
%STRING = $VIEW('CCAJRNL'|'CCAJLOG'|'CHKPOINT'|'CHKPNTS')
```

$VIEW returns a string that describes the usage of the specified stream with the following character format:

Describes	LENGTH	Displays
In use DDNAME	8	Alphanumeric characters
Percentage used	4	Integer with percentage sign
Maximum size	9	Number of tracks
Current size	9	Number of tracks
Unused spaces	2	Number of tracks
VOLSER	6	Alphanumeric characters
Separating back slash	1	A back slash
Dataset name	Variable	Dataset name
Separating back slash	1	A back slash
Generation number	Variable	Generation number with a leading plus sign

You must ensure that the %variable being used to hold the requested information is defined as STRING LEN 80 or greater, for example:

```
%STRING IS STRING LEN 80

BEGIN
%STRING = $VIEW('CCAJRNL')
PRINT %STRING
END

CCAJRNL  0%  37  0  0SU000\OHIO1.ONLINE.JOURNAL
```

MSGCTL command enhancement

On a message by message basis, use the MSGCTL command to change message processing options to more precisely accommodate the needs of your site and applications.

The hierarchical structure of MSGCTL is reinstated and message types—ER, RK, AD, and MS—are mutually exclusive. Each processing option is handled independently of other options, as each option is designed to manage a single function.

Note: The parameter, CUSTOM (7) option is now obsolete.

In addition, the new message processing mechanism eliminates significant processing for all but the MSGCTL changed messages. These changes result in decreased CPU consumption.

New or updated MSGCTL command options

NOACTION option

The NOACTION option suppresses almost all error message processing. When an error is generated, Model 204 checks an indicator flag to determine if that error message needs to be processed and whether the processing options have been changed via MSGCTL processing. If an error is found to have no processing requirements—that is, NOACTION—processing returns immediately to the next instruction. NOACTION has no effect on the following messages:

- Counted messages that occur during compilation
- Initialization, start up, and restart
- Termination

The decreased number of instructions necessary to process the NOACTION option versus the NOAUDIT option results in a large CPU processing reduction.

Caution: If you allow User Language programs to continue processing past an error that has been suppressed with the NOACTION option of the MSGCTL command, you must expect abends, incorrect database results, recovery errors, the inability to recover or other unpredictable consequences. CCA strongly recommends that you consider carefully the circumstances, before you introduce the NOACTION option.

AUDIT*xx* options

These options can be used for any message type—AUDITER, AUDITRK, AUDITAD, and AUDITMS—to change the current message type to another *xx* type message and its related processing options.

The NOAUDIT option functions for a message of any type, suppressing the auditing of the designated message.

NOAUDITxx options

The NOAUDITxx options function only for a message of that designated *xx* type.

Option	Changes message type	To message type
NOAUDITER	ER	RK
NOAUDITRK	RK	AD
NOAUDITAD	AD	MS
NOAUDITMS	MS	NOAUDIT

Independent option assignment

For example, message M204.1030: INVALID COMMAND is designated to be processed as both a counting error and an ER type message.

If you wish to turn off auditing and journaling for this message, issue the following command:

```
MSGCTL M204.1030 NOAUDIT
```

The message will no longer be written to the journal or the audit trail. However, it remains a counting error and appears on the user's terminal.

To change this message so that it is also not a counting error, issue the following command:

```
MSGCTL M204.1030 NOAUDIT NOCOUNT
```

Again, the message will no longer be written to the journal or audit trail, and now it is not counted toward the ERMX limit. However, it still appears on the user's terminal.

Caution: CCA strongly recommends that you do *not* apply the NOCOUNT option to messages that are part of User Language compilation processing. Counting errors typically terminate the compilation process. In instances where you make compilation errors NOCOUNT, compilation is not terminated. Failure to stop compilation can result in invalid information being made available to the evaluation process resulting in subsequent snaps.

SWITCH STREAM command

The SWITCH STREAM command switches a stream to the next member of a parallel, ring, concatenated, or Generation Data Group (GDG) stream. You can issue the SWITCH STREAM command for the following streams: CCAJLOG, CCAJRNL, CHKPOINT, or CHKPNTS.

SWITCH STREAM processing

When the SWITCH STREAM command is issued, the following messages are produced:

```
M204.2712: STREAM streamname IS BEING SWITCHED
M204.2712: STREAM streamname SWITCHED VIA COMMAND
```

- When switching a journal stream, CCAJRNL or CCAJLOG, the currently active dataset in that stream is closed. The next dataset defined to the stream is opened when the next write to that stream is required.
- When switching a checkpoint stream CHKPOINT or CHKPNTS, the currently active dataset in that stream is closed after the next record is written to that dataset.

If not in an extended quiesce, CCA recommends that you follow a SWITCH STREAM CCAJRNL command with a CHECKPOINT command to ensure that you have a checkpoint in the current journal.

The SWITCH STREAM command may be useful for CCAJLOG, but is rarely necessary for CHKPOINT or CHKPNTS.

SWITCH STREAM CCAJRNL during extended quiesce

In previous releases, if you used the checkpoint quiesce feature, a switch to the next journal member at checkpoint quiesce could occur only when CCAJRNL was defined as a ring stream.

However, a switch at checkpoint quiesce may be very useful for all stream configurations of CCAJRNL: ring, parallel, concatenated, and GDG. The switch marks the point where the CCAJRNL data collected thus far is not needed in subsequent REGENERATE processing for files backed up during the quiesce. If file backups or dumps are taken during the checkpoint quiesce, only CCAJRNL data collected after the checkpoint quiesce is needed for subsequent REGENERATE processing of those files.

A checkpoint is automatically taken, if you issue a SWITCH STREAM CCAJRNL command while you are in extended quiesce, as shown in Table 1-3.

Table 1-3. SWITCH STREAM command during an extended quiesce

Step	Command issued	Purpose
1	CHECKPOINT SET EXTENDED QUIESCE	Enables extended quiesce
2	CHECKPOINT or automated checkpoint	The next checkpoint, automated or command initiated, begins the extended quiesce When the checkpoint is successful, the extended quiesce is entered. While in extended quiesce, you cannot issue a CHECKPOINT command. However, you can issue a SWITCH STREAM command, such as SWITCH STREAM CCAJRNL.
3	SWITCH STREAM CCAJRNL	When the checkpoint is successful, extended quiesce processing can begin for backups, SnapShots, or any activity that does not involve updating.
4	CHECKPOINT END EXTENDED QUIESCE	Concludes extended quiesce. Should recovery be required due to a failure during extended quiesce, journals created prior to the SWITCH STREAM command will not be required, as the last checkpoint resides in the current CCAJRNL member.

This automated checkpoint functionality of SWITCH STREAM CCAJRNL applies only during checkpoint quiesce. The automated checkpoint functionality is not supported for CCAJLOG, CHKPOINT, or CHKPNTS.

SWITCH STREAM limitations

In order for a stream to be switched, there must be a target dataset to switch to. If there is no target dataset, the following message is issued:

```
M204.2712: MEMBER membername IS INELIGIBLE FOR SWITCHING
```

Consider the following example:

```
DEFINE STREAM CCAJRNL WITH SCOPE=SYSTEM PARALLEL=(JRNL1,JRNL2) MINAVAIL=2
DEFINE DATASET JRNL1  WITH SCOPE=SYSTEM DSN=CCAJRNL.JRNL1 OLD
DEFINE STREAM  JRNL2  WITH SCOPE=SYSTEM GDG=J2 CONTROL=J2CTL
DEFINE DATASET J2     WITH SCOPE=SYSTME DSN=CCAJRNL.GDGBASE.JRNL2 CATALOG -
                      GEN=+1 CYL PRI 500
DEFINE DATASET J2CTL  WITH SCOPE=SYSTEM DSN=CCAJRNL.GDGBASE.JRNL2.CTL OLD
```

Since dataset JRNL1 has no target dataset for a switch, when the JRNL1 dataset is marked full, the number of available parallel stream members (MINAVAIL) drops to one, since only JRNL2 now has space available. Since the number of available members is now less than MINAVAIL the Online would stop with a CCAJRNL full message.

If before JRNL1 fills, a SWITCH STREAM CCAJRNL command were issued, the following messages would be produced:

```
M204.2712: MEMBER JRNL1 IS INELIGIBLE FOR SWITCHING
M204.2712: STREAM JRNL2 IS BEING SWITCHED
M204.2712: STREAM CCAJRNL - NOT ALL MEMBERS SWITCHED
```

Member JRNL1 was not switched. However, if JRNL1 has not filled, the parallel stream will remain open because the number of available members (MINAVAIL) is still 2. Stream JRNL2 always has a target dataset to switch to since it is a GDG. Nevertheless, whenever JRNL1 fills, the minimum available members will be less than MINAVAIL and the stream will be closed and the run will terminate.

Because stream members are not locked before switch processing, and because all members are not switched at exactly the same time, the number of records in individual datasets of a parallel stream may not be identical. This is of no consequence to recovery or REGENERATE, but should be noted. However, the total number of records in each member is identical.

Previously, you had to concatenate all journals into one dataset or specify a concatenated CCAGEN DD statement.

SWITCH STREAM command for concatenated streams

The same limitation exists for the last member in a concatenated stream. Since there is no additional member to switch to, the following message is issued:

```
M204.2712: STREAM streamname IS INELIGIBLE FOR SWITCHING
```

If all members of a stream are switched, the following message is issued:

```
M204.2712: STREAM CCAJRNL SWITCHED VIA COMMAND
```

Handling streams without records

You cannot switch a stream member that contains no records. So, in the previous case, if the first G1 stream member has just become full and the newly opened (second) GDG member contains zero records, then a SWITCH STREAM CCAJRNL command is not processed.

Since J1 is ineligible, and G1 (second member) is empty, no switch occurs. The messages issued are:

```
M204.2712: MEMBER J1 IS INELIGIBLE FOR SWITCHING
```

```
M204.2712: MEMBER G1 IS EMPTY AND CANNOT BE SWITCHED
M204.2712: SWITCH WAS UNSUCCESSFUL
```

Implementing the date/time stamp feature

A date/time stamp (DTS) feature has been added to Model 204, so that you can easily discern when a record in a file was last changed. The date/time stamp feature is supported for ONLINE and BATCH204.

- Your site may customize:
  - The date/time stamp field name

    There is no default date/time stamp field name as delivered. You can define the date/time stamp field name in the CUST source code or on User 0's parameter line by setting the new DTSFN parameter.
  - The field definition

    A user with file manager privileges can define the date/time stamp field for each file you want enabled, and subsequently, you must set the FOPT parameter to include X'10'. Please note, the FOPT=X'10' value cannot be set until the date/time stamp field has been defined to the file.
  - The data that is stored in this field

    The default date/time stamp format combines the system DATE and TIME fields. If your site prefers a different field value or a different data format, see "Customizing the date/time stamp field name in CUST source code" on page 1-34 and "Activate the date/time stamp feature" on page 1-35.
- The DTS feature is implemented at the file level. In other words, the decision to include a file in DTS processing is made on a file-by-file basis.
- Except for DELETE RECORD(S) processing, all other updates are captured in BATCH204 and ONLINE enabled jobs.
- Precision and accuracy of the date/time stamp is to the millisecond when you use the date/time stamp as shipped. CCA recommends that the date/time stamp field be defined as ORDERED CHARACTER to support ordered index FIND processing. The length of this field is 17 bytes.
- The date/time stamp field is updated at the beginning of commit processing when a new record is stored or an existing record is updated—change, insert, or delete field.
- You can read and display the date/time stamp as any other field. User Language may be used to find and display date/time stamp field data.
- You can exclude a user thread from date/time stamp field maintenance by resetting the DTS parameter to 0.

The date/time stamp routine executes once for each transaction if the user thread has set DTS=1 and the files being updated have an FOPT value that includes X'10'. The data built by the DTS routine updates the field specified by the DTSFN parameter.

Handling the date/time stamp field in a file

The DELETE FIELD command is prohibited for the DTSFN field in a file when the FOPT=X'10' is set. Attempting to do so results in the following message:

```
M204.2727: CAN'T DELETE DTS FIELD WHEN FOPT=X'10' IS ON
```

Installing the date/time stamp feature

If you select the date/time stamp (DTS) feature as an installation option, the DTSR and CUST object code will be linked into the ONLINE and BATCH204 modules. See the Model 204 installation manual for complete instructions.

Customizing the date/time stamp field name in CUST source code

The DTSFN field is established by the DTSINIT routine in the CUST object code. As shipped, this routine does not provide a default field name. You may customize the CUST source code to establish a date/time stamp field name. Edit DTSINIT to change the following line:

```
CUSTFN  DC  0C
```

to the desired field name, for example,

```
CUSTFN  DC  C'MY DATE TIME STAMP FIELD NAME CONTAINING
SPACES'
```

The maximum length for a field name is 255 characters.

Default instructions in the routine move the system DATE and TIME into the field in the format:

```
CCYYMMDDHHMMSSMMM
```

Table 1-4. Date/time stamp format

Where	Specifies	For example: 20080315163510345
CC	Century	20 for 21st century
YY	Year	08 for 2008
MM	Month	03 for March
DD	Date	15 for the fifteenth day of the month
HH	Hour on a 24-hour clock	16:00 hours or 4:00 PM
MM	Minutes past the hour	35
SS	Seconds past the minutes	10
MMM	Milliseconds	345 past the seconds

Making modifications to the default date/time stamp value

If the default data that is supplied in the DTSUB routine in CUST is not what you want or require, the source code for the DTS field value is available for modification. This field area of up to 255 bytes may be completely formatted as your site desires.

Assembler statements to build the data in Table 1-4 that comprise the date/time stamp field values are placed in the DTSUB entry point in the CUST module. This entry point contains instructions that build a field value in the area pointed to by T3 (CCA register notation) upon input. Register T1 contains packed decimal date in form of 0CYYDDDF and register T2 contains time of day in milliseconds since midnight.

- You are not required to use the *ccyymmddhhmmssmmm* format for your DTS file date. You may change the format of the date and time data to suit your site requirements.
- You may customize the date/time stamp field to include flags or other system data you deem necessary.
- You could add a suffix to the default data, such as USERID or ACCOUNT.
- The default format and data content is DISPLAY (up to 255 bytes long), however, you could use Binary or Float instead.
- There is no requirement for any position to have a special meaning to Model 204 internal routines.

Note: Each site is responsible for whatever changes they make to the DTSUB entry point. CCA does not support changes beyond those documented in “Installing the date/time stamp feature”.

Activate the date/time stamp feature

After you link DTSR and CUST into your modules, your jobs will run with date/time stamps active as the default. This is because the user parameter DTS defaults to a value of 1 which means:

```
DATE TIME STAMP ACTIVE FOR THIS USER
```

If the new modules are not linked into the Online, then any attempt to reset the DTSFN, DTS, or FOPT=X’10’ parameters will result in the following message:

```
M204.2740: INVALID PARAMETER parameter: DATE TIME STAMP
SUPPORT NOT LINKED IN
```

SUBSYSMGMT enhancements

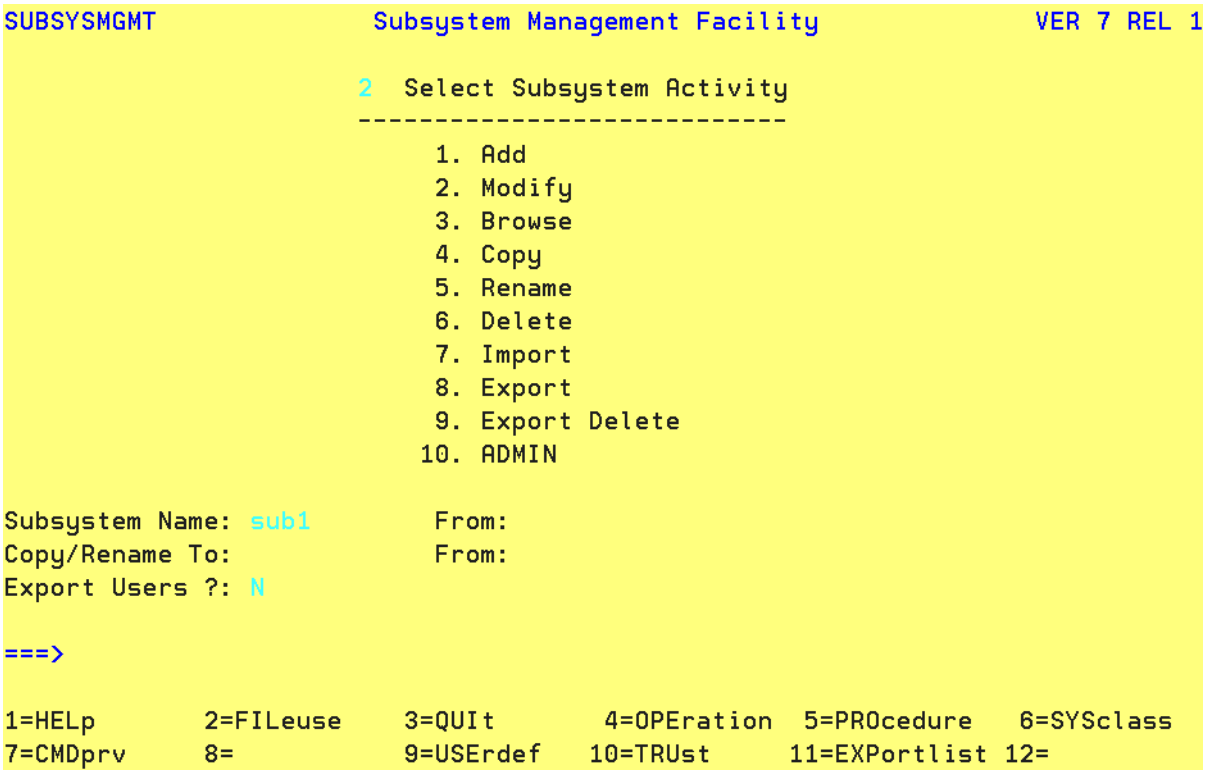
The SUBSYSMGMT facility lets you manage and maintain subsystem definitions. This enhancement to the SUBSYSMGMT facility lets you easily maintain the privileges associated with the subsystem commands.

Formerly, the SUBSYSMGMT facility let you change the command privileges for an individual subsystem class in an individual subsystem definition. This enhancement lets you use the SUBSYSMGMT facility to change the command privileges for a set of subsystem classes in a set of subsystem definitions.

To use this feature, you must have privileges to use the SUBSYSMGMT facility as well as the privileges to update the individual subsystems.

Revising command privileges

Using the main screen of the SUBSYSMGMT facility you can enter a pattern in the Subsystem Name field and select PFkey 7 to change the command privileges for a set of subsystems. Another screen appears where you can enter a pattern for a subsystem class along with the new command privileges for the START, STOP, TEST, DEBUG, RESUME, SUSPEND, REFRESH commands. You can also obtain a list of all of the subsystem classes and subsystem definitions that fit the pattern criteria.



To update the command privileges, select the 2. MODIFY option on the main menu, enter a subsystem name or pattern in the Subsystem Name field and press PF7. The COMMAND PRIVILEGES screen is displayed next. However:

- If you enter a pattern in the name field, but press another function key, the pattern is treated as an individual name, not a pattern. PF7 is only valid with the MODIFY option.
- If PF7 is pressed, but the MODIFY option was not specified, then the following error message is displayed:

  ```
  SUMnnn: PFkey or command only valid with Modify option.
  ```

- If the Subsystem Name is left blank, the following error message is displayed:

  ```
  SUMnnn: Subsystem name or pattern is required.
  ```

Introducing the Subsystem Privileges screen

```
SUBSYSMGMT                    COMMAND PRIVILEGES

< 11  > of < 20  >       ------------ Change all selected: ------------
Subsystem      Class     Start Stop  Test  Debug Resume Suspend Refresh
                         (Y/N) (Y/N) (Y/N) (Y/N)  (Y/N)  (Y/N)   (Y/N)
SUB1           *
------------  ---------  ----- ----- ----- -----  -----  -----   -----
x SUB1         SC19         N     N     N     N      N      N       N
x SUB1         SC2          N     N     N     N      N      N       N
x SUB1         SC3          N     N     N     N      N      N       N
x SUB1         SC4          N     N     N     N      N      N       N
x SUB1         SC5          N     N     N     N      N      N       N
x SUB1         SC6          N     N     N     N      N      N       N
x SUB1         SC7          N     N     N     N      N      N       N
x SUB1         SC8          N     N     N     N      N      N       N
x SUB1         SC9          N     N     N     N      N      N       N
x SUB1         USERS        N     N     N     N      N      N       N

===>

1=HELp        2=           3=QUIt    4=LISt   5=DESelect all     6=
7=BACkward    8=FORWARD    9=       10=      11=UPDate priv     12=END
```

After your screen entries are read, a list of subsystem names and subsystem classes based on the pattern criteria is displayed. The pattern criteria used for SUBSYSTEM NAME is the value that was specified on the main screen. The pattern criteria for SUBSYSTEM CLASS defaults to '*" when the screen is initially read.

If any of the subsystems in the list are enqueued by other users, then the command privileges for the subsystem cannot be updated. In this case, 6=DISplay appears on the screen to display the list of subsystems that are enqueued.

Choosing other classes or names

If you wish to refine the list of Subsystem Classes or Subsystem Names then you may enter a new pattern in the Subsystem Name or Subsystem Class field and press PF4 to view the new list.

The maximum number of subsystem classes that may be processed at one time is 1000. If the list exceeds 1000, then the following error message is displayed:

```
SUM096: Refine criteria: # of subsystems classes, exceeds
max(1000).
```

Entering changes in the COMMAND PRIVILEGES screen

To update the command privileges for a particular subsystem class, the command privilege must be set to 'Y' or 'N' in one or more of the subsystem command privilege fields. In addition, the Subsystem Class must be selected by placing an 'x' in front of the subsystem name and the class name. You can also update more than one subsystem class at a time by specifying a command privilege of one of the 'Change All Selected' columns. If the Subsystem name and class are not selected, then the command privileges are not updated for that particular subsystem class.

At this point, the user presses PF11/UPDate or PF12/END to update the command privileges for the entire set of selected subsystem classes. If a record enqueuing conflict occurs during the update process then only a partial update can be performed. If this should occur, then the user can press PF6/DISplay to display the list of subsystem classes that are enqueued and cannot be updated at this time. The user also has the option of pressing PF11/UPDate or PF12/END a second time, to apply the partial update or press PF3/QUIt to not apply the update.

Command privileges are in effect immediately after they are updated and can be changed if the subsystem is active or not.

You may also deselect particular subsystems or classes on the list by deleting the X that precedes the subsystem or class name.

On Table 1-5, the list of PF key options and commands are alternatives that accomplish the same thing. (Commands are used with the Enter key.) Each PF key function listed can also be performed by adding 12 to that PF key and using the resulting PF key. Commands shown may be abbreviated to the first three characters, which are capitalized.

The procedure performs one of the following operations based on the PF key pressed, as listed in Table on page 1-39.

Table 1-5. PF keys

PF Key	Command	Purpose
1	HELp	Display help information on this screen.
3	QUIt	Exit from Command Privileges Screen return to main menu for SUBSYSMGMT.
4	LISt	Displays a list of subsystems and classes based on the pattern specification in the Subsystem Name and Class fields.
5	DESelect all SELect all	De-select all subsystem names on the list. Select all subsystem names on the list. Toggles between DESelect and SELect.
6	DISplay	Displays a list of subsystems or classes that are enqueued by another user. The command privileges for the subsystems and classes on the list cannot be updated at this time.
7	BACkward	Scroll up on subsystem names list
8	FORward	Scroll down on subsystem names list
11	UPDate	Update all subsystem classes with new command privilege. Only the command privileges that are changed are updated.
12	END	Return to the main menu after updating the command privileges that are specified. If a particular command privilege is left blank then it is not updated.

Command privileges can be changed regardless of whether a subsystem is active or not. The new privileges are in effect immediately once they are changed.

SUBSYSMGMT command privileges error messages

See “SUBSYSMGMT error messages” on page 7-39.

Sending e-mail messages via User Language

The User Language $SNDMAIL function simplifies sending an e-mail message to issuing a single function call. Now you can easily send an e-mail message using User Language without writing a procedure and without having an extensive knowledge of sockets or SMTP protocol.

Limitations of the $SNDMAIL function

The $SNDMAIL function works as documented in this section on z/OS systems.

Under z/VM, $SNDMAIL is supported with the exception that the SMTPCADD and SMTPSADD addresses must be supplied in internet protocol (IP) address protocol, not as domain names. Likewise, if the client and server names are specified using the SMAILD image rather than using the new CCAIN parameters SMTPCADD and SMTPSADD, the SERVERDDNAME and CLIENTDDNAME values must be supplied rather than the SERVERNAME and CLIENTNAME values.

Although Model 204 supports the Internet Protocol (IP) addressing IPv4, it does not yet support IPv6.

Introducing $SNDMAIL

The new $SNDMAIL function sends an e-mail message using the values you enter for the $SNDMAIL parameters. The $SNDMAIL function lets you send an e-mail message with an optional binary attachment.

Syntax

The syntax for the $SNDMAIL function is:

```
$SNDMAIL(%SUBJ,%RCPT,%BODY,%FROM,%TO,%CC,%BCC,'EMAILD',%BINARY)
```

Where:

- %SUBJ, which is optional, specifies a %variable or literal that contains the topic line.
- %RCPT, which is optional, specifies a %variable or an image that contains the destination addresses, separated by commas.
  - If %RCPT is omitted, the destination addresses are taken from the %TO, %CC and %BCC parameters.
  - If %RCPT is an image, the image name must be enclosed within quotation marks, for example 'DEPT-ADDR'.
- %BODY specifies a %variable or an image that contains the text of the message. If %BODY is an image, the image name must be enclosed in quotation marks, for example 'MSG-TXT'.
- %FROM, which is optional, specifies a %variable that contains the FROM header field of up to 255 bytes. If %FROM is omitted, the logged-on user ID

is used. The e-mail server may append the domain name to the given %FROM value.

- %TO, which is optional, specifies a %variable that contains the TO header field of up to 255 bytes. Address must be enclosed in carets (<>), for example: CCA-help <maintinfo@cca-int.com>.
- %CC, which is optional, specifies a %variable that contains the CC (copy) header field of up to 255 bytes. Address must be enclosed in carets (<>), for example: CCA-help <maintinfo@cca-int.com>.
- %BCC, which is optional, specifies a %variable that contains the BCC (blind copy) header field of up to 255 bytes. Address must be enclosed in carets (<>), for example: CCA-help <maintinfo@cca-int.com>.
- 'EMAILD', which is optional, specifies image data. If 'EMAILD' is present, it must be in the following format:

```
IMAGE EMAILD
  SERVERPORT   IS BINARY LEN 4
                            /? SMTP port number, usually 25 ?/
  CLIENTPORT   IS BINARY LEN 4
                            /? Port used by this Online     ?/
  SERVERNAME   IS STRING LEN 255
                            /? SMTP server name             ?/
  CLIENTNAME   IS STRING LEN 255
                            /? Name of client               ?/
  SERVERDDNAME IS STRING LEN 15
                            /? SMTP server IP address       ?/
  CLIENTDDNAME IS STRING LEN 15
                            /? Name of client IP address    ?/
END IMAGE
```

If 'EMAILD' is omitted, the default values that are defined by CCAIN startup parameters—SMTPCADD, SMTPCPRT, SMTPSADD, and SMTPSPRT—are used.

- %BINARY, which is optional, specifies a %variable or literal with the name of a binary attachment that will be appended to the message. The binary object is assumed to be in the Universal Buffer at offset 0 (also called position 1). The length of the attachment is defined by the $BUFFER_USED value. If the %BINARY parameter is omitted, there is no attachment.

  The function encodes the binary attachment using Base64 encoding.

Setting CCAIN parameters for $SNDMAIL

You can set the following CCAIN parameters as default values for $SNDMAIL.

- SMTPCADD: SMTP client address

SMTPCADD is the default client address, the TCP/IP domain for the operating system, for the $SNDMAIL function. You can specify SMTPCADD as:

– Domain name of up to 255 bytes

– If the first character of the address is an alphabetic character (A to Z), it is automatically evaluated as a domain name.

– Internet Protocol (IP) address, IPv4 format

You can override SMTPCADD for a specific call by entering a different value in the EMAILD parameter of the $SNDMAIL function call.

- SMTPCPRT: SMTP client port

  SMTPCPRT is the default client port for the $SNDMAIL function. The value can be any value in the user range—normally above 4095 and up to 32767.

  You can override SMTPCPRT for a specific call by entering a different value in the EMAILD parameter of the $SNDMAIL function call.

- SMTPSADD: SMTP server address

  SMTPSADD is the default server address for the $SNDMAIL function. You can specify SMTPSADD as:

  – Domain name of up to 255 bytes

  – If the first character of the address is an alphabetic character (A to Z), it is automatically evaluated as a domain name.

  – Internet Protocol (IP) address, IPv4 format

  You can override SMTPCADD for a specific call by entering a different value in the EMAILD parameter of the $SNDMAIL function call.

- SMTPSPRT: SMTP server port

  SMTPSPRT is the default server port for the $SNDMAIL function. SMTPSPRT is normally set to the standard SMTP value of 25. You can override SMTPSPRT for a specific call by entering a different value in the EMAILD parameter of the $SNDMAIL function.

Error codes for $SNDMAIL

The $SNDMAIL function returns a return code. Many of the codes are existing $SOCKET function codes. However, in the case of a non-zero (error) return code, the following message is issued to the user:

```
M204.2799: $SNDMAIL ERROR errortext
```

When an attempt is made to set or reset the SMTPCADD or SMTPSADD parameters and the address translation fails, Model 204 issues the following messages:

```
RESET SMTPCADD CCAGGGG
```

```
*** M204.2834: ADDRESS LOOKUP FAILED FOR CCAGGGG
*** 1  M204.1123: PARAMETER SMTPCADD NOT RESET
```

Examples of running $SNDMAIL

Example 1

```
%FROM = 'me@mydomain.com'
%TO = 'My Friend <you@yourdomain.com>'
%BNAME = 'mypicture.jpg'
MODIFY BUFFER CLEAR
 IN LOBSVR FOR 1 RECORD WHERE SVRKEY = %SVRKEY
  %ISIZE = $LOBLEN(SVRLOB)
  PRINT 'SIZE OF ' WITH %SVRKEY WITH ' IS ' WITH %ISIZE
   BUFFER, 1, %ISIZE = SVRLOB, 1, %ISIZE
 END FOR
PRINT 'BUFFER USED IS ' WITH $BUFFER_USED
%SUBJECT = 'THIS SENDS A JPG FROM LOBSVR - ' WITH %SVRKEY
%BODY      = 'THIS BODY CONTAINS ' WITH %SVRKEY
%RC = $SNDMAIL(%SUBJECT,,%BODY,%FROM,%TO,,,,%BNAME)
```

In the previous example, a binary object (BLOB) is retrieved from the database and placed at position 1 in the buffer. The $SNDMAIL function sends this as a Base64-encoded attachment.

Note: The second parameter (the RCPTTO addresses) is omitted. The function creates this using the %TO parameter.

Example 2

```
%TO IS STRING LEN 30
%TO   = 'JANE <YOU@YOURCORP.COM>'
%BODY = 'See you at 9.30 pm today in the gym'
%RC   = $SNDMAIL(,,%BODY,,%TO)
```

Note: The length of %TO was defined LEN 30 prior to entering the value of the e-mail address, because the default of 20 would truncate the %TO value, making the e-mail note undeliverable, although the $SNDMAIL function return code is 0.

Example 3

* The EMAILD parameter is used to override the default domain names and ports.

```
IMAGE EMAILD
     SERVERPORT  IS BINARY LEN 4
```

```
     CLIENTPORT  IS BINARY LEN 4
     SERVERNAME  IS STRING LEN 255
     CLIENTNAME  IS STRING LEN 255
     SERVERDD    IS STRING LEN 15
     HOSTDD      IS STRING LEN 15
  END IMAGE

  PREPARE IMAGE EMAILD
%EMAILD:SERVERNAME  = 'MAIL.SERVER.COM'
%EMAILD:CLIENTNAME  = 'MYMVSSYSTEM'
%EMAILD:SERVERPORT  = 25
%EMAILD:CLIENTPORT  = 5601
%FROM     = 'ME@MYCORP.COM'
%TO       = 'JANE <YOU@YOURCORP.COM>'
* RCPTTO is defined as an image here, allowing the string of
* addressees to be more than 255 bytes long.
  IMAGE RCPTTO
     EMRCPT1    IS STRING LEN 255
  END IMAGE
  PREPARE IMAGE RCPTTO
%RCPTTO:EMRCPT1    = 'YOU@YOURCORP.COM'
IMAGE BODY
     BODY1      IS STRING LEN 240
     BODY2      IS STRING LEN 240
  END IMAGE

  PREPARE IMAGE BODY
%S = 'EXAMPLE 1'                    /?    SUBJECT LINE     ?/
%BODY:BODY1 = 'THIS IS BODY1 WHICH IS PART OF AN IMAGE'
%BODY:BODY2 = 'SO THE BODY TEXT CAN BE MORE THAN 255 BYTES LONG'
%RC        = $SNDMAIL(%S,'RCPTTO','BODY',%FROM,%TO,,,'EMAILD')
```

ASCII translation

All text, except an optional binary attachment, is translated from EBCDIC to ASCII before sending.

Line ends

Text that is part of the header and body can include EBCDIC CRLF strings (X'0D25') that are translated to ASCII along with the rest of the text.

Reserved names

The $SNDMAIL function uses Interface names CCAEM000 through CCAEM999. The user does specify these names—the link name is generated by the function.

Socket limits

Up to 31 sockets are available on each link. If no more sockets are available on a particular link (CCAEM000, for example), the next link is tried. Thus, a maximum of 31,000 simultaneous e-mail threads is theoretically possible.

Size of e-mail message

The FROM, TO, CC, BCC, RCPTTO addresses, and body length, excluding the optional binary attachment, must total less than 27454 bytes.

Return code

The $SNDMAIL function returns a fixed fullword binary return code (see “Handling error condition return codes” on page 1-46 for possible values).

RCPTTO parameter

If the $SNDMAIL EMAILD parameter is omitted, the RCPTTO addresses are constructed from the addresses contained in the TO, CC, and BCC variables.

The sum of the lengths of the RCPTTO address string, (if supplied) and the body text cannot exceed 30,700.

Using quotation marks

You must enclose the names of images within quotation marks, for example: ‘BODY’. However, do not enclose the percent variable values within quotation marks, for example: %SUBJECT, not ‘%SUBJECT’.

Server and client names in the EMAILD parameters

In the EMAILD image, you can specify the server and/or client name as a domain name, for example SMTPSERVER.YOURDOMAIN.COM, or as an Internet Protocol (IP) address. If the domain name is specified, a TCP/IP call looks up the IP address.

- If one field is present, all fields must be present, even if the optional fields are left blank (see the example below).
- If a name or port field is absent, it is taken from the new CCAIN parameters SMTPSADD, SMTPCADD, SMTPSPRT, and SMTPCPRT.
  - The two port parameters have defaults: the SMTPSPRT default is 25; the SMTPCPRT default is 7700.
  - The two address parameters, SMTPSADD and SMTPCADD, can be specified either as domain names of up to 255 bytes or as IP addresses. If the address begins with an alphabetic character (A to Z), it is automatically evaluated as a domain name.

The four new CCAIN default parameters SMTPSADD, SMTPCADD, SMTPSPRT, and SMTPCPRT can be examined using VIEW and changed using RESET.

Security considerations

None.

References

For more information about the standards used in this function, please refer to these Internet *Request for Comments* documents:

- RFC822, Standard for the format of ARPA Internet Text Messages
- RFC2821, Simple Mail Transfer Protocol
- FRC3548, The Base16, Base32, and Base64 Data Encoding

Handling error condition return codes

These are the possible Sockets return codes.

SMTP Error text	Numeric value	Meaning
ERBADARG	-999	BAD ARGUMENT
ERNOTOPN	-998	NOT OPEN
ERNOSTOR	-997	NO STORAGE
ERIMAGER	-996	IMAGE FAILURE
ERBADLNK	-995	BAD LINK
ERNOTREL	-994	SOCD NOT RELEASED
ERPHYERR	-993	LINK DEAD
ERNOLINK	-992	LINK UNKNOWN
ERTRUNC	-991	VARIABLE TRUNCATION (READ CALL)
ERNOSOCK	-990	SOCKET NOT FOUND
ERTRUNC2	-989	TRUNCATION ERROR ON IMAGE
EROPSYS	-988	FUNCTION UNSUPPORTED ON THIS OPSYS
	-987	Not used
ERNOSRCD	-986	NO SRCD
ERFDSLIM	-985	TOO MANY SOCKETS

ERBADUBN	-984	UNIVERSAL BUFFER NOT THERE
ERBADUBL	-983	UNIVERSAL BUFFER LENGTH ERROR
ERBADUBA	-982	UNIVERSAL BUFFER ALLOCATION ERROR
EREMREAD	-981	ERROR WITH EMAIL SERVER READ
EREMWRIT	-980	ERROR WITH EMAIL SERVER WRITE
ERGHBERR	-979	SKIP=0 ERROR ON GETHOST CALL
ERGHERR1	-978	T1 LT 0 ERROR ON GETHOST CALL
ERGHERR2	-977	NO ADDRESSES RETURNED

When an error message (with code 5xx) is returned from the SMTP server, the following error message is displayed:

```
M204.2796: 'SMTP %C'
```

where %C is the error message returned by SMTP.

For the other error codes above, the following error message is returned, in addition to $STATUSD being set.

```
M204.2799: '$SNDMAIL ERROR %C'
```

where %C is listed in the following table:

%C in message 2799	**$STATUSD code(s)**
IN PARAMETER	999
- SOCKET NOT FOUND	990
- FUNCTION NOT SUPPORTED ON THIS SYSTEM	998
- NO SRCD	986
- TOO MANY SOCKETS	985
WITH UNIVERSAL BUFFER	984, 983, 982
Reading from SMTP server	981
Writing to SMTP server	980
Using GETHOSTBYNAME lookup	979, 978, 977
GETTING STORAGE	997
IN IMAGE PROCESSING	996
WITH TCP LINK	995, 998, 993, 992
- SOCD NOT RELEASED	994
READING DATA - TRUNCATION	991

Additional features

MAXSIMIO to save IOS Branch Entry storage

During initialization, Model 204 calculates a default number of Disk Buffer I/O control blocks (DBIDs). If you are using IOS Branch Entry, this default number may be unnecessarily high. Use the MAXSIMIO parameter to allocate a smaller number of DBIDs. (See “MAXSIMIO: Maximum number of simultaneous disk I/Os” on page 6-15.)

Model 204 uses the following formula to calculate, as the default, the maximum number of DBIDs that would be needed:

```
32 * (NSUBTKS + NSERVS)
```

Using a lower MAXSIMIO value, when you are also using IOS Branch Entry, limits the number of DBIDs allocated and results in storage savings. If you are not using IOS Branch Entry, an explicit MAXSIMIO setting is ignored and the value calculated by Model 204 is used instead.

FILEORG bit X’80’ to improve nonpreallocated field extraction

CPU savings now result from the improved field extraction implemented using a field scanning and skipping process.

Include the X’80’ bit for FILEORG during file creation. All nonpreallocated (or non-OCCURS) fields, with the exception of FLOAT, CODED, and BINARY fields, have always been stored with a preceding length byte. With FILEORG bit X’80’, this length byte is now added to every nonpreallocated field, including FLOAT, CODED and BINARY fields. This eliminates several instructions when processing field scans and, therefore, speeds up the field extraction process. The CPU savings in this enhancement are most noticeable when you have built records with very many fields.

Affects of field type on file sizing

- If a field was defined as type STRING, there is no affect on these fields.
- If a field is defined as a BINARY or CODED field, the sizing goes from 4+2 (4 for the value, 2 for the field code) bytes to 4+1+2.

Note: In some instances, the length byte allows Model 204 to do simple compression of BINARY, CODED, and FLOAT values. As an example, for FLOAT values, compression simply strips trailing zeros. FLOAT representations of integer values often have multiple trailing zeros, thus compress well. For those cases where zero is a common value in a FLOAT field, with FILEORG bit X’80’ the field would have a length of 4 (1+1+2=value, length byte, field code) as opposed to a length of 10 (8+2=value, field code) for a FLOAT len 8 field.

Statistics in V7R1.0

The following statistics are new in V7R1.0. They include the statistics for MP lock waits—MPLKPREM and MPLKWTIM.

Table 1-6. Statistics new in V7R1.0

Statistic	Counts	Type
MPLKPREM	Total elapsed time in milliseconds, across the main task and all subtasks, the Online spent waiting due to operating system preemption. This is the elapsed time between when an MP lock becomes available (lock post) making a task ready to run, and when the task actually gets the CPU. That preemption delay is caused by the operating system dispatching other tasks ahead of this task. Only collected in MP/204 systems.	System
MPLKWTIM	Total elapsed time in milliseconds, across the main task and all subtasks, the Online spent waiting for MP locks. Only collected in MP/204 systems.	System
LKPTIME	Lock preemption time—Total elapsed time in milliseconds, this task spent waiting, due to operating system preemption. This is the elapsed time between when an MP lock becomes available (lock post) making the task ready to run, and when the task actually gets the CPU. That preemption delay is caused by the operating system dispatching other tasks ahead of this task.	MONITOR TASK command

Support for larger statistics

To accommodate the sites that need to run Model 204 around the clock for many days at a time, the size for most Model 204 statistics has been increased from single to double word. See *Appendix A: Statistics in 64-bit Architecture.*

FBMX statistic

The FBMX statistic has been withdrawn. It is no longer calculated and it is not available for auditing.

MOVE statistic

With multiple KOMMS, the MOVE statistic now represents the number of times Model 204 switched from one user to another. In earlier releases it specified the number of times a particular user was moved to a server.

Improved snap management

The snap formatter is more robust, capable of recovering first-level as well as second-level errors without disabling the Online. The errors in question are program exceptions in unexpected places. For example, when the snap formatter assumes a pointer is valid, but it is not. This avoids a 4095 ABEND, followed by a RECURSIVE ENTRY TO ESTAE, which would bring down the Online.

The improvement increases the usability of CCASNAPs in the presence of push down list errors. Also, additional information has been added to the snaps, such as user and system statistics.

The new parameters, SNAPFAIL and SNAPFLIM, identify snap failures that have happened and set a limit on the number of failed snaps. (See “SNAPFAIL: Number of CCASNAP failures” on page 6-21 and “SNAPFLIM: Snap failure limit” on page 6-21.)

At the completion of recovery from a failed snap, one of the following M204.1449 messages is sent to the operator and CCAJRNL or CCAJLOG, as well as saved in the VIEW ERRORS table.

- The following error message increases the value of SNAPFAIL by one:

  ```
  M204.1449: ERROR WHILE PROCESSING CCASNAP
  ```

- The following error message indicated that snap formatting has been disabled, because the value of SNAPFAIL equals the value of SNAPFLIM.

  ```
  M204.1449: ERROR WHILE PROCESSING CCASNAP, CCASNAPS
  DISABLED
  ```

Both versions of message 1449 sets the Online and Batch return codes to 100.

Disk buffer overrun detection

To better detect disk buffer overruns a boundary indicator, or fence, has been added between each buffer in the disk buffer pool. This fence is a string of 8 bytes of hexadecimal FF. As a result the size of the disk buffer pool has increased by (NUMBUF*8) bytes.

New SYSOPT2 options

SYSOPT2=X’40’

Record sets—found sets, including FDWOL found sets, sorted sets, lists, and LPU lists—are traced through entries in the record locking table. One entry is required for each segment (49,152 records) in the record set. These entries are CCATEMP page numbers.

When SYSOPT2=X’40’, the entries contain 4-byte CCATEMP page numbers. Setting SYSOPT2=X’40’ provides a substantial increase in the number of simultaneous record sets that can be concurrently active in a given Model 204 run. Therefore, if you set SYSOPT2=X’40’, you should also at least double LRETBL.

- When the SYSOPT2 setting *does not* include X’40’, then at any given time the bit maps corresponding to all users holding found sets of any kind must fit into CCATEMP pages designated as the small model page pool no matter how large CCATEMP has been allocated.
- When the SYSOPT2 setting *does* include X’40’, the CCATEMP page restriction is removed and user found sets can be placed anywhere within CCATEMP. This includes both the small model page pool and the CCATEMP expansion area, allowing for the possibility of a greater number of concurrent found sets being held by all users.

SYSOPT2=X’80’

- With SYSOPT2 X'80' *off,* all dynamic allocations default to use the existing TIOT option unless the XTIOT option is specified on a DEFINE or ALLOCATE command for the dataset.
- With SYSOPT2 X'80' *on,* all dynamic allocations that specify the OLD and DIRECT file options use the XTIOT option, unless the TIOT option is specified on a DEFINE or ALLOCATE command for the dataset. To use this setting XMEMOPT=X’02’ must also be set.

With the use of dynamic allocation and the XTIOT option, only the amount of processor storage limits the number of allocated datasets.

Interrupting a COPY STREAM command

The COPY STREAM command now recognizes a bump or an urgent message and terminates processing. The following messages are issued:

```
M204.0303: USER INTERRUPTED, PROCESS TERMINATED

M204.1859: xxxx BLOCKS HAVE BEEN COPIED TO OUTPUT
STREAM/DATASET
```

- BUMP processing causes a user restart with all messages written to the audit trail.
- A message issued by either the BROADCAST URGENT or WARN commands is received by the user following the M204:0303 and M204.1859 messages. BROADCAST URGENT or WARN does not restart the user.

Large dataset support

Large datasets are now supported for Model 204 database files and journal files, including CCAJLOG, GDG (z/OS only) and stream-based journals, and sequential datasets, including file dumps.

The Model 204 DEFINE DATASET and ALLOCATE DATASET commands include the new argument, LARGE, that indicates a dataset may have more than 64K tracks.

z/OS datasets may be SMS or non-SMS managed. A z/OS or z/VM dataset must be sequential (defined with the PS option) to qualify as a large sequential dataset.

Model 204 files may be dumped to or restored from large sequential datasets. Utilities for journal processing—UTILJ, MERGEJ, Audit204—support large format journal files as input datasets. MERGEJ also supports large datasets as output.

All options of the M204UTIL z/VM utility have been updated to allocate and erase large format datasets and to initialize, write and read non-indexed OS-format VTOCs on large volume DASD. See the *Model 204 z/VM Installation Guide* for information on preparing and using OS format disks under VM.

EXCP and IOS Branch Entry support is also provided.

Compilation error message management

The initial User Language compilation error can cause a cascade of subsequent error messages that do not help you analyze the problem. Furthermore, the error may not point out just where the problem occurred. The new COMPERR parameter that lets you determine how many error messages—one or many—to display and to highlight the code that caused the error.

You can set COMPERR to X’01’, X’02’, X’04’, or a sum of the options you choose.

- COMPERR=X’01’ points to (>> <<) the first error and displays only that error.

```
** 1***** M204.2775: INVALID EXPRESSION IN INITIAL CLAUSE ************
    ***** %A IS STRING LEN 10 DP 0 INITIAL(>>C'A1B2'<<) STATIC *******
    ***** (FILE = ULP42, PROCEDURE = V4R2.TEST.42FINIT.ERRORA, LINE=10
    ***** M204.1042: COMPILATION ERRORS
```

- COMPERR=X’02’ underlines (__) the error and displays only that error.

```
** 1***** M204.2775: INVALID EXPRESSION IN INITIAL CLAUSE ************
    ***** %A IS STRING LEN 10 DP 0 INITIAL(C'A1B2') STATIC ***********
    ***** (FILE = ULP42, PROCEDURE = V4R2.TEST.42FINIT.ERRORA, LINE=10
```

```
***** M204.1042: COMPILATION ERRORS
```

- COMPERR=X'03' both underlines and points to (>>__<<) the error and displays only that error.
- COMPERR=X'04' displays all messages.
- COMPERR=X'07' both underlines and points to (>>__<<) the error and displays all messages.

MONITOR command synonym

The synonym M is now supported for all forms of the MONITOR command, for example:

- M DISKBUFF is equivalent to MONITOR DISKBUFF
- M VTAM is equivalent to MONITOR VTAM
- M SERV is equivalent to MONITOR SERV

Checkpoint configuration support

You may define the CHKPOINT and CHKPNTS datasets using parallel streams. The only restriction on CHKPOINT and CHKPNTS stream definitions is that they may not contain CMS datasets. There are no restrictions on the type of stream you can use to define the CCAJRNL and CCAJLOG journals.

ERRMSGL - Setting the length of saved error messages

The new ERRMSGL parameter provides the ability to set the number of bytes stored for saved error messages--messages returned by $ERRMSG and $FSTERR. Prior to this release, only the first 79 characters of an error message were saved. ERRMSGL can be set to any value from 80 to 256—that length includes a count byte. The value is rounded up to an 8-byte multiple. For example, if you set ERRMSGL=99, it will be rounded to 104, that value is then reduced by 1 for the count byte, thus allowing up to 103 characters of an error message to be saved. See "ERMX: Maximum number of errors for user's session" on page 6-9.

Server size requirements for saved error messages

Increasing ERRMSGL increases the requirement of the fixed table size of a server. This may necessitate an increase in your SERVSIZE settings. The size requirement for ERRMSGL is:

```
3 * (ERRMSGL - 80)
```

For example, increasing ERRMSGL to its maximum of 256 would increase the fixed server requirement by (3 * (256-80) -1) or 527 bytes. Setting ERRMSGL

higher can be useful for saving long error messages otherwise lost due to truncation. The value of ERRMSGL can be displayed with the VIEW command or the $VIEW function in User Language.

For a full update to the SERVSIZE formula, see “Calculating fixed table size” on page 1-68

IP address using TN3270 connection to VTAM session: IPADDR

TN3270 connections via Telnet can be made to a Model 204 job with IODEV=7 connections defined in the job. For such a connection, it may be desirable to know the IP address of that user’s VTAM session. This information is now available via the IPADDR parameter. You can retrieve the IP address using the:

- VIEW command

  ```
  VIEW IPADDR
  ```

- User Language $VIEW function

  ```
  %IPADDR=$VIEW('IPADDR')
  ```

Running REGENERATE with the IGNORE argument

Using the new IGNORE argument with a REGENERATE command bypasses the file parameter list (FPL) update timestamps. This allows you to run REGENERATE processing with one CCAGEN at a time, instead of requiring a single concatenated journal. If necessary, you can run single CCAGEN datasets one at a time in multiple REGENERATE steps.

For example, the first REGENERATE command may be as follows and provide the first journal:

```
REGENERATE FILE abc FROM dumpabc
```

A subsequent run may provide the second journal and specify:

```
REGENERATE FILE abc IGNORE
```

The second run picks up from where the first REGENERATE processing ended and applies the second journal updates.

Previously, you had to concatenate all journals into one dataset or specify using a concatenated CCAGEN DD statement.

Caution: If you omit a journal, it is not reported. Therefore, use the new option with care.

The new IGNORE keyword is not valid with the FROM option. See “REGENERATE: Recovering a file in two passes” on page 6-56 for syntax details.

Enhanced user scheduling

The system manager uses the PRIORITY command and the PRIORITY parameter on a user IODEV line in CCAIN to allocate Model 204 resources to users based upon their relative service requirements. Users can be in one of three priority classes: LOW, STANDARD, or HIGH. In general, HIGH priority users receive service sooner than STANDARD priority users and STANDARD priority users receive service sooner than LOW priority users.

Refining priority management

In V7R1.0 this scheduling mechanism has been enhanced. V7R1.0 priority management effectively doubles the current dispatching priority range from 1-127 to 0-253 and provides for more fine tuning of user priorities.

PRIORITY command and PRIORITY parameter syntax enhancements

The PRIORITY command has been enhanced to support priority ranges. The original formats, LOW, STANDARD, and HIGH, will continue to be supported using their default ranges.

PRIORITY command syntax

```
PRIORITY [userno [,LOW|STANDARD|HIGH]]
```

or, introduced in V7R1.0

```
PRIORITY [userno [,cur|,(cur,min,max)] [,keyword=value]]
```

PRIORITY parameter syntax

```
PRIORITY=[LOW | STANDARD | HIGH]
```

or, introduced in V7R1.0

```
PRIORITY=(cur,min,max)
```

where:

- *userno* specifies the user number to modify or display. If no other parameters are specified, only the user is displayed.
- *cur* specifies a new, current priority, 0-253, for the specified user.

  A user's current priority is not required to be within this range. However, as the user ages, the current value will rise or fall until it falls within the given range.
- *min* specifies a new, minimum priority, 1-253, for the specified user.
- *max* specifies a new, maximum priority, 1-253, for the specified user.
- *keyword=value* specifies the value to be assigned to the specified argument for this user. To reset values to system defaults, specify a null value, for example: IOSLICE=0 is a valid value.

The following keywords are recognized:

Command keyword	Parameter keyword	Specifies
IOSLICE	UIONSLIC	CPU milliseconds allowed while user is I/O bound.
CPUSLICE	UCPUSLIC	CPU milliseconds allowed while user is CPU bound.
SLCWAIT	USLCWAIT	Sleep time in milliseconds, invoked each time a user reaches minimum priority level.
SLCMAX	USLCMAX	Number of Stop-Loop-Checks (SLCs) before CSLICE invoked (max=65535). Reducing this number increases the accuracy of the slice interval, ***However, CPU overhead increases.*** **Note:** SLC is an internal Model 204 routine that prevents infinite user loops.

Duration of PRIORITY assignments

Once a priority has been assigned, that priority remains in effect until it is changed by another PRIORITY command or until you log out of Model 204. On logoff or restart, all priority parameters are reset to either their user default values (set on IODEV) or their system values.

Setting PRIORITY=0

If a user priority is set to zero, that user will no longer be dispatched. Instead, the user remains logged in, but is suspended during evaluation. The suspension can occur at command level or at the bottom of a FOR loop. That user must be reset to a non-zero priority to continue. (Exception: If the user is updating or holds critical resources, they will be allowed to run to COMMIT or END of request before being suspended.

Examples

The following priority command changes User 38's current priority to 100, minimum to 80 and maximum to 120. The value of IOSLICE, for this user only, is also changed to 60.

```
PRIORITY 38,(100,80,120),IOSLICE=60
```

To reset argument values back to system defaults, specify a null value. For example to change User 38 back to the system default value for IOSLICE:

```
PRIORITY 38,,IOSLICE=,
```

The trailing commas is required and indicated the null value. The double commas after 38 indicate that current priority or priority range has been omitted.

Delaying work

When necessary, the system administrator can delay work to accommodate other, higher priority work. For example, if low importance BATCH2, HORIZON or other IODEV threads are running and work of higher importance must be run the low importance threads may be set to an extremely low and narrow priority range and SLCWAIT and SLCMAX values may be added.

Let's say you want to drastically increase the elapsed time a BATCH2 thread requires to run, essentially run it as a very low priority, background task. If the BATCH2 thread is user 59, the following PRIORITY command will allow that thread to run once every 2 seconds (SLCWAIT) for 100 milliseconds (IOSLICE). After 100 milliseconds or when the thread issues any kind of wait for an external event—disk I/O, READ SCREEN, server swap, record/resource locking conflict resolution, pause, and so on, the thread will wait for two seconds be fore being dispatched again:

```
PRIORITY 59,(5,5,5),IOSLICE=100,SLCMAX=1,SLCWAIT=2000
```

Users may also be suspended with the PRIORITY command. If you suspect a runaway application, but want to confirm before bumping the user, you could suspend that user by setting priority to zero:

```
PRIORITY 72,0
```

Note: Setting a user to low or zero priority, however, must be done with care. Record locks continue to be held for a LOW or zero priority user. Other users, who need to process those records, may be blocked.

PRIORITY command output

The output of the PRIORITY command includes a header line which indicates the meanings of the statistics in the user lines that follow. User and server numbers occupy up to five characters.

If PRIORITY is entered with no parameters, all users are displayed. You may abbreviate the PRIORITY command to PRI.

```
PRIORITY

USER USERID   P CUR,MIN-MAX   SLICE  IOSLICE CPUSLIC MAX SLCWAIT SERV     CPU
   0 NO USERI S 253,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.001
   1 BECKETT  S 061,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.013
   2 LESTER   H 127,080-127   0.000I   0.070   0.100  50   0.00     5    0.170
   3 MATSUZAK S 061,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.012
   4 PENNY    H 104,080-127   0.000I   0.070   0.100  50   0.00     1    0.422
   5 WAKEFIEL H 114,080-127   0.000I   0.070   0.100  50   0.00     3    0.105
   6 VARITEK  S 079,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.063
   7 YOUKILIS S 079,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.069
   8 PEDROIA  S 079,032-079   0.000I   0.070   0.100  50   0.00     6    0.133
   9 LOWELL   S 072,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.032
  11 LUGO     S 079,032-079   0.000I   0.070   0.100  50   0.00     2    0.112
```

The priority (P) column (third from the left in the previous priority display) may have the following values:

Priority column	Specifies…
L	Low
S	Standard
H	High
*	User defined
Z	Sleeping (priority=zero);
?	Deferred priority change in progress, which will take effect when the user issues COMMIT command, BACKOUT command, or request processing ends.)

The SLICE column displays either I for I/O bound or C for CPU bound.

Consider this scenario of user scheduling

For a user, who in this example is USER11, USERID=LUGO, PRIORITY can be set as follows:

```
PRIORITY 11,(20,16,40),SLCWAIT=2000,SLCMAX=2,      X
IOSLICE=30,CPUSLICE=10
```

USER11 will begin work with a priority of 20. Any users with STANDARD priority or higher will receive CPU cycles ahead of USER11. Once USER11 has received 30 milliseconds of CPU (the IOSLICE value at which the user is declared CPU-BOUND), USER11 will be declared CPU-BOUND for the next 10 milliseconds.

At selected times during the processing (loop, FIND, FOR, and so on), the Model 204, internal, Stop-Look-Check (SLC) routine will be called on behalf of USER11 to evaluate his time slice. Since SLCMAX=2, the first SLC will skip the check of the time slice. On the second call, the usage will be examined. If USER11 has exceeded 10 milliseconds (CPUSLICE), then USER11 will be time sliced (rescheduled) and a user with higher priority is allowed to run.

Each time USER11 is rescheduled and is found to be CPU-BOUND, his priority will be lowered by 2. So, after approximately 40 (30+10) milliseconds of CPU, USER11 now has a priority of 18. USER11 continues to run and two SLC calls later, USER11 is again time sliced.

USER11 has now consumed approximately 50 (30+10+10) milliseconds of CPU and his priority is now 16. This is the bottom of his range, so the SLCWAIT parameter becomes active.

Since SLCWAIT=2000, USER11 is placed in a timed 2-second (2000 milliseconds) wait and will not be rescheduled for evaluation until this time has elapsed. If no other work is found for Model 204 to process, Model 204 will enter

an operating system wait until USER11 or another user becomes eligible to be dispatched. In the meantime, while Model 204 waits, the operating system may schedule other address spaces, virtual machines, or partitions.

New CUSTOM setting

The CUSTOM=(9) setting suppresses all output from all forms of the PRIORITY command.

CCATEMP statistics changes

The M204:2622 message is printed twice at termination and has been changed to report the high water mark of the number of pages used in both the small model area and the expansion area.

```
M204.2622: HWM CCATEMP PAGES USED IN SMALL MODEL AREA
           = nnnnnn, MAX AVAILABLE = 65536

M204.2622: HWM CCATEMP PAGES USED IN EXPANSION AREA
           = nnnnnn, MAX AVAILABLE = nnnnnn
```

The two CCATEMP statistics, TEMPHIE and TEMPHIS have been changed to report the high water mark of pages used in the CCATEMP expansion area and in the small model page pool, respectively. Previously, these parameters were the highest page number used in the expansion area and in the small model page pool. Since highest page number used is not helpful in determining how close each of these areas is to becoming full, the high water mark is now reported.

MERGEJ support for journal types

The MERGEJ utility can determine the journal type being merged. If the dataset type is unknown, MERGEJ sets a return code of 2 and issues the following warning message.

```
*** WARNING: UNKNOWN DATASET TYPE: PLEASE SPECIFY TYPE VIA
PARAMETER SETTING OF JRNL, JLOG OR BOTH.
```

A merged journal may contain audit information (CCAJLOG) and/or recovery (CCAJRNL) information. A dataset containing only:

- Recovery information cannot be used with AUDIT204.
- A dataset containing only messages and audit information cannot be used for REGENERATE or RESTART.

If the MERGEJ utility attempts to use CCAJRNL or CCAJLOG datasets that are not the first dataset for each DD respectively (such as, GDG members for CCAJRNL that are the fourth, sixth, and so on, datasets), the initial record is not present and the type of data, JRNL or JLOG, cannot be determined. The

MERGEJ utility terminates with a return code of 2 indicating unknown dataset type.

To inform the MERGEJ utility as to the type of data to be processed set the PARM parameter to JRNL, JLOG, or BOTH. For example, in z/OS specify:

```
EXEC PGM=MERGEJ,PARM=BOTH
```

ENQCTL command enhanced

The ENQCTL command has been enhanced to list all entries in the shared DASD enqueue list that resides in a file's File Parameter List (FPL). Previously, a maximum of eight entries was listed.

Formerly, when a file could not be opened due to shared DASD enqueueing conflicts, a maximum of eight entries was listed using the M204.0582 message number. With the enhancement to the ENQCTL command, the M204.0582 message is issued as many times as necessary.

SQL outer join features expanded

SQL join expressions with join types LEFT, RIGHT, CROSS, and UNION are supported where any table can be a base or a nested table or a viewed table.

Usage Notes

- In a query, a Left Outer Join and Right Outer Join can be combined in any order.
- In a view definition, a Left Outer Join and Right Outer Join can be combined in any order.
- A nested table can be used in any place where a base table can be used.
- A view can be used in any place where a base table can be used.
- The ON clause can contain any columns from the joined table.
- A subquery can contain a subquery.

In the following example, Join can be either a Left or Right join.

SQL table expression

```
T1 Join T2 ON Pred1 Join T3 ON Pred2 Join T4 ON Pred3
```

Model 204 evaluates the previous code in the following sequence of steps:

1. Table T1 joined with table T2 by ON Pred1
2. The expression (T1 Join T2 ON Pred1) is joined with T3 by ON Pred2
3. The expression ((T1 Join T2 ON Pred1) Join T3 ON Pred2) is joined with T4 by ON Pred3

SQL SELECT extended

A 1,000 column limit for SQL SELECT statements has been established.

Controlling MQ subtask release

You can use the MQSUBREL parameter to control whether the MQ subtask is released at the end of request processing in subsystems with AUTOCOMMIT=N.

You can set the MQSUBREL parameter on User 0's CCAIN line to affect all users. Or, you can reset it for an individual user to affect only that specific user. At login or a user restart, the MQSUBREL parameter is returned to the value specified by the User 0 CCAIN parameter line.

- A setting of 0 lets you retain the subtask and preserve pre-V7R1.0 behavior. Choose this setting when you expect the same user to use MQ services soon after ending a request.
- A setting of 1 lets you release the subtask for another user. Choose this setting when you expect that other users may be waiting for a free MQ subtask.

AUDIT204 and UTILJ reports now include header

AUDIT204 and UTILJ reports now have a 2-line header, as shown in the following examples, that reflect the circumstances of your site.

```
AUDIT204 UTILITY - VERSION 7.10D,
DATE/TIME OF RUN: 03/31/2009 11:31:38

UTILJ UTILITY - VERSION 7.1.0D
DATE/TIME OF RUN: 03/31/2009 00:18:48
```

Setting the FICREATE parameter

If you create a file without setting ESIZE or XSIZE, then FICREATE is set at 6, specifying Model 204 V6R1.0. Files with FICREATE=6 can be opened by Model 204 V6R1.0 or later.

If you create a file with non-zero XSIZE, but no ESIZE, then FICREATE is set at 7. Files with FICREATE=7 cannot be opened by Model 204 V6R1.0 or earlier.

If you create a file with non-zero ESIZE, with or without a non-zero XSIZE, then FICREATE is set at 8, specifying Model 204 V7R1.0.

RESTART command changes and enhancements

RESTART ROLL BACK processing now requires RCVOPT=8, even if ROLL FORWARD is not specified. Roll back only to a sub-transaction checkpoint is now available, so this type of recovery requires journaling of information for subsequent secondary recovery, it needed. In addition, GDGs are now supported for CHKPOINT during recovery, which allows for unlimited size recovery checkpoints.

If at CHKPOINT open during RESTART ROLL BACK processing there is no CHKPOINT defined for the job, the job displays the following message and waits for a response from the operator.

```
OO ///  UNABLE TO OPEN CHKPOINT, REPLY RETRY OR CANCEL
```

Messages consolidated into *Model 204 Messages Manual*

In addition to new messages written for V7R1.0 features, the *Model 204 Messages Manual* includes all messages generated by Model 204 and related products. Messages formerly located in other manuals in the Model 204 documentation set have been consolidated into one manual and no longer appear in their former location.

- “Chapter 9: Prefixed Messages” now includes FLOD Exit error messages and M204HASH error messages that were moved from the *Model 204 File Manager’s Guide.*
- “Chapter 10: VTAM Messages” were moved from *Model 204 VTAM Printer Support Summary*.
- “Chapter 11: Connect★ Error Code Messages” were moved from *Connect★ Suite Programming and Installation*.
- “Chapter 12: Dictionary/204 Messages” were moved from *Model 204 Dictionary/204 and Data Administration Guide*.

Furthermore, “Appendix B: Messages Grouped by Return Code” was created.

Managing DCBLRECL to avoid an ABEND

If a BATCH2 job sends a line to CCAOUT that is longer than the DCBLRECL defined for CCAOUT, the line will be truncated at DCBLRECL-4 length to avoid an S002-18 abend. This only occurs for RECFM=V datasets. The following message will be issued for each line that exceeds the DCBLRECL length:

```
M204ULIF.0009: PREVIOUS CCAOUT LINE LONGER THAN DCBLRECL-
4; LINE TRUNCATED
```

To avoid the error, either RESET OUTCCC to a value less than DCBLRECL-4 or increase DCBLRECL for CCAOUT.

Identifying a mismatch between a process and a REMOTEID

Improve messages so that it is clearer why a CNOS connection is rejected by the server.

You could have an inbound request for an existing process, although the process does not seem (according to the definitions) to have any logical connection with the REMOTEID, the VTAM connection over which the connection request has come. Model 204 issues the following message:

```
M204.2837: NO LOGICAL CONNECTION BETWEEN PROCESS %C AND
REMOTEID %C
```

ERMX parameter

If ERMX is exceeded due to User Language compilation errors, the user is no longer restarted.

Page intentionally left blank

Upward compatibility issues

Version related issues

Once a Model 204 file is dumped using V7R1.0, it is not backward compatible with previous releases, only forward compatible.

- If a Model 204 file is dumped using V7R1.0, you must use V7R1.0 to restore the file.
- If a Model 204 file is dumped using a prior release, you may restore it using the previous release or V7R1.0.

Table E size greater than zero

Files created in V7R1.0 with Table E size greater than zero are not backward compatible and cannot be opened in earlier releases.

Date/time stamp files

You cannot use the date/time stamp feature with files created before V6R1.0. You must recreate the files in V6R1.0 or later. See *Chapter 3: Date/Time Stamp Support* for more details.

Processing V6R1.0 files

Files created with Table X or FILEORG bit X'80' cannot be opened in Model 204 V6R1.0 or earlier. However, files created in later versions without a Table X or FILEORG bit X'80' can be opened and processed in Model 204 V6R1.0.

Version-specific journals

Because the journal file format has changed in V7R1.0, journals created by Model 204 prior to V7R1.0 cannot be processed in V7R1.0. The reverse is also true: journals created in V7R1.0 cannot be processed by an earlier version of Model 204.

Version-specific SVC reminder

If you use SVCs, you must install new SVCs for each new release of Model 204. CCA does not support transferring SVCs from a previous release into a newer release. Be sure to update M204XSVC and XDM.

Communicating between versions of Parallel Query Option

PQO V7R1.0 cannot communicate with PQO V6R1.0 or earlier. If you attempt this, Model 204 issues the following message:

```
M204.2327: PQO VERSION INCOMPATIBILITY WITH REMOTE NODE:
location
```

Moving to 64-bit support

Increase in LPDLST

User push down list (PDL) requirements have increased, because 64-bit registers are now saved in the PDL, instead of 32-bit registers. Also, all PDL frames are double word aligned, so some of them are padded with an extra 32-bit word.

Going to 64-bit support increased the space requirements for the user pushdown list. The minimum value for the LPDLST parameter has been increased from 2176 to 5000.

MQMD Version 2

MQ messages written with the MQ Message Descriptor (MQMD) Version 2 cannot be read by V6R1.0, or earlier, Onlines. However, MQ messages written with MQMD Version 1 can by read by V7R1.0 Onlines.

The MQMD is a part of the MQ message. It is referred to as the message header. An Online that uses Version 1 of the MQMD header, such as MQ/204 V6R1.0, cannot read messages from a Version 2 MQMD header, due to additional fields in the message header. However, an Online that uses Version 2 of the MQMD header, such as MQ/204 V7R1.0, can read messages from a Version 1 MQMD header, because the Version 2 header new fields provide default values.

Retiring in V7R1.0

KOMMOPT parameter retires

The KOMMOPT parameter is obsolete. The default is that all multi-user jobs have a KOMM for each user (or multiple KOMMs), which is the equivalent of a KOMMOPT=1 setting. This change should lead to a decrease in server size requirements.

DDLUTIL support retired

Mainframe support for DDLUTIL has been removed in this release of Model 204. See the *Connect ✦ Suite Installation and Programming Guide* to populate the SQL catalog from your PC.

z/VM dataspace no longer supported

The APSYPAGE parameter and the DSPOPT=X’01’ are not supported in Model 204 for the z/VM operating system.

BAUTOTYP parameter is obsolete

The BAUTOTYP parameter may be viewed on files created prior to V7R1.0, but not reset.

COMMIT in FOR and FOR RECORD NUMBER (FRN) loops changed

User Language COMMIT statements inside of FOR and FOR RECORD NUMBER loops now release only the transaction back out (TBO) lock, meaning lock pending update locks, not single record locks. All other single record locks remain, such as, the single record share lock associated with a FOR RECORD NUMBER statement and a single record exclusive lock associated with updating User Language statements while the FOR or FOR RECORD NUMBER loop is active.

Formerly, User Language COMMIT statement inside of FOR or FOR RECORD NUMBER loops released all locks on the record.

Updates to TBO and non-TBO files in the same transaction

In V6R1.0 and earlier, requests that attempted to update a TBO and non-TBO file would not be compiled. Now such requests are compiled. This enhancement means that you can update a non-TBO file, commit it, and then update a TBO file—or reverse file order—in the same request.

Requests that attempt to update TBO and non-TBO files without an intervening COMMIT statement will compile, but will fail during evaluation with the following message:

```
***  1  CANCELLING REQUEST; M204.2771: ATTEMPT TO UPDATE
TBO AND NON-TBO FILES IN THE SAME TRANSACTION
```

Table requirements increased

Increase in STBL for MQ/204 sites

The increased STBL requirement is because MQ control blocks are kept in STBL, and they have increased in size by several hundred bytes. The increase depends on how many MQPUT and MQGET statements are compiled in any one transaction, so it will be the high water mark for the largest User Language compilations.

Increased VTBL requirements

An extra 32 bytes of VTBL is needed for each FIND criteria in the FIND statement with the largest number of criteria. For example, if your request has a FIND statement with six criteria, the VTBL requirements would increase as follows:

```
6 FIND statement criteria * 32 = 192 bytes
```

If you sized VTBL at precisely what is needed to compile in an earlier release, compilation may fail in this release with an error.

LVTBL requirements

The VTBL allocation for an ordered index cursor has increased by 12 bytes. Ordered indexed cursors are used for statements such as:

```
FOR EACH VALUE of an ORDERED field
```

Calculating fixed table size

Use the following formula to calculate fixed table size, the FIXSIZE parameter value:

```
Fixed table size = 2520

      + ((LAUDPROC + 9) * 4)dwr

      + (LIBUFF + 4)

      + (LOBUFF + 5)dwr

      + (LOUTPB)dwr

      + ((NGROUP + 12) * (NRMTFILE + NFILES + 1))

      + ((NORQS*3) + 2)dwr + (NRMTFILE + 1))dwr

      + (3 * (ERRMSGL - 80))
```

Each term of this formula that is followed by *dwr* must be double word rounded to the next multiple of eight. For example, if the value of LOBUFF is 500, then the term (LOBUFF + 5) = 505 must be rounded to 512, the next multiple of 8.

If SYSOPT = 1 or 2 (indicating CCASYS or CCAGRP), add 1 to the value of NFILES used in the formula. If SYSOPT = 3 (indicating both CCASYS and CCAGRP), add 2.

If any SQL threads are specified in CCAIN (IODEVs 13, 17, or 19), add 6712 bytes for C language work areas.

Table 1-7 shows the minimum, maximum, and default values for parameters that affect fixed server table sizing. The rightmost columns show the relevant units of measure; for example, the maximum value of NORQS is 32767 entries (not bytes). The values of LIBUFF and LOBUFF may need to be increased for

SQL processing. Recommended values are LIBUFF=3000 and LOBUFF=5000.

Table 1-7. Fixed server table values

Parameter	Default	Max	Bytes/entries
LAUDPROC	21	253	Bytes
LIBUFF	255	32767	Bytes
LOBUFF	256	32767	Bytes
LOUTPB	0	3000	Bytes
NFILES	2	16383	Entries
NGROUP	5	16383	Entries
NORQS	5	32767	Entries
NRMTFILE	0	16383	Entries
ERRMSGL	80	256	Bytes

Parallel Query Option and Large Object data

The Parallel Query Option does not support access to actual Large Object data—neither BLOBs nor CLOBs. For PRINT ALL INFORMATION statements of a remote record, the Table B portion of the Large Object field is displayed, but not the data itself. In this way, you can identify remote records that have Large Object fields, although you will need to access the file locally to work with the Large Object data.

Changed commands

TABLEB RECLEN command output

The output displayed by a TABLEB RECLEN command is dependent on whether Table X is defined for the file.

- If Table X is defined for the file, the TABLEB RECLEN command displays the estimated average length of the *base* records in the file.
- If Table X is not defined for the file, the TABLEB RECLEN command displays the estimated average length of the *logical* records in the file, as in earlier versions of Model 204. A logical record is a base record, plus any extension records.

MSGCTL processing

MSGCTL processing is completely rewritten for V7R1.0. The CUSTOM (7) option is obsolete and no backward compatibility is provided to support prior MSGCTL processing.

You must review any MSGCTL commands you have embedded in job streams, User Language procedures, or APSY subsystems to ensure they are updated to work as needed.

The new implementation provides options to provide more direct and less complicated user management of message processing. Some of the changes include:

Message option	V7R1.0 behavior
COUNT	No longer automatically changes the message type to ER.
NOCOUNT	No longer changes the current message type.
AUDITxx	Use these options to change a message type
NOACTION	Used to suppress message processing

For full details on all MSGCTL updates see “MSGCTL command enhancement” on page 1-26.

Note: This new implementation requires a reassembly of the user message module (MSGU). This may also include third-party software messages that you install.

Parameter changes

OPSYS parameter change

In V7R1.0, the view-only OPSYS parameter, which returns the value for the operating system under which Model 204 is running, changed the meaning of the X’000100’ value. Instead of representing Hitachi HITAX VOS3, as it did in V5R1, it now represents z/OS 1.7 or later.

New default value for CFRLOOK

The CFRLOOK parameter has a new default, CFRLOOK = 1.

CUSTOM=7 withdrawn

Changes in the MSGCTL command options made the parameter setting CUSTOM=7 obsolete.

In addition, the hierarchical structure of messages is reinstated. If the NOAUDITER option is issued in a MSGCTL command for an ER message, the message is lowered to a message type RK. The NOAUDITRK option lowers an RK message to an AD message. The NOAUDITAD option lowers an AD message to an MS message. Finally, NOAUDITMS lowers an MS message to the status of not being audited (NOAUDIT).

A NOAUDITxx option processes only a message of the 'xx' type. However, NOAUDIT works on any message type. In the same way, AUDITxx may be used on any message type and results in the message becoming type 'xx'.

See “MSGCTL: Setting message output” on page 6-48 for the specific changes in syntax and additional changes in behavior.

CUSTOM=(9)

CUSTOM=(9) added to suppress output from all forms of the PRIORITY command.

DEFINCBP renamed DEFINCP

The DEFINCP parameter (formerly DEFINCBP) indicates the default percentage increase for Tables B, D and X. For more details, see “DEFINCP: Default automatic Tables B, D, and X size percentage increase” on page 6-6.

LAUDPROC parameter rounded up

The LAUDPROC parameter is always rounded up based on the following formula:

```
LAUDPROC = (LAUDPROC + 19 rounded to double word
boundary)-19
```

For example, if LAUDPROC is set to 25, then the value of LAUDPROC is rounded to 29.

SCHDOPT parameter setting X’10’

SCHDOPT=10 means that CSLICE will always verify that the long request values are not exceeded. This will result in increased CPU overhead. If this bit is zero, the default, pre-V7R1.0 CSLICE behavior is maintained, meaning that long request values are checked only if the current user exceeds its CPU slice limits (CPUSLICE) without an intervening SWAIT.

Supporting TPROCESS COBOL applications

You must relink any CICS TPROCESS applications using AMODE=31 as AMODE=24 is not supported.

Changes in statistics in V7R1.0

Improved FBWT statistic

The FBWT statistic that reports the number of waits for a disk buffer was under reporting. The calculation has been corrected to report more accurately. You

may see a larger number in the FBWT statistic than you are accustomed to in your reports.

Changes in return codes in V7R1.0

If a CRAM error is detected, message M204.0336 is written and the Online is terminated. Likewise, if a VTAM open fails, message M204.0199 is issued and the Online is terminated. In prior releases, the Online return code remained zero. In V7R1.0, the return code will be 4 in both Online and Batch.

Return codes expanded

UTILJ now sets five different return codes.

Code	Message condition
0	Normal return
16	OPEN failed for CCAPRINT
20	OPEN failed for CCAJRNL
24	I/O error occurred reading or writing CCAJRNL
28	CCAJRNL contains one or more corrupted journal blocks

Return code 28 is issued whenever a corrupted journal block is detected while reading CCAJRNL. The following message is also printed at the end of LISTING:

```
**** THIS CCAJRNL CONTAINS ONE OR MORE CORRUPTED JOURNAL
BLOCKS - PLEASE CONTACT CCA CUSTOMER SUPPORT
```

This is a serious error that must be brought to the immediate attention of CCA.

Setting user SVCs correctly

The following message is issued whenever the SMFSVC is not set to a value greater than 199 and less than 256. IBM documentation requires that all user SVCs be in the range 200 - 255 inclusive.

```
M204.2789: SMFSVC = nnn INVALID; SMF ACCOUNTING DISABLED
```

Buffer size increases

Eight bytes were added to the end of every buffer, above and below the bar, to detect buffer overruns. The new buffer size per page is 6192 bytes (or 6184 plus 8). For memory constrained systems, this could lead to the loss of a few buffers, particularly below the bar.

LPDLST minimum and default value change

The minimum size of the User Push Down List (PDL) is now set to 5000 bytes. LPDLST cannot be reset to a value less than 5000. You may need to increase the size of your servers (SERVSIZE), and possibly server datasets, to allow for the increased PDL usage.

If you attempt to set LPDLST parameter to a value less than 5000, either in a CCAIN input stream or via a UTABLE command, the following message is issued.

```
M204.1149: LPDLST HAS BEEN SET TO ITS MINIMUM VALUE: 5000
```

2

File Management Enhancements

In this chapter

- Overview
- Storing base records and extension records in separate tables
- Using Table X for extension record storage
- Sizing and using Tables B and X
- COMPACTB data compaction enhancements
- Automatic increase for Tables B, D, and X
- COMPACTE data compaction for Table E

Overview

This document contains information pertinent to Model 204 V7R1.0 tables. This chapter also introduces Table X for extension records and the automatic increase for Tables B, D, and X.

You can reclaim space in Table E using the Table E compactor and the TABLEE command.

Storing base records and extension records in separate tables

A Model 204 file can be created in V7R1.0 to allow the maximum storage of 16 million base records in Table B even when extension records are in use. A *logical record* consists of a base record that has the same record number as the *base record*, and zero or more extension records. Each extension record has a unique internal record number. The base record points to the first extension that points to the next one.

Prior to V7R1.0, each extension record resided on a separate Table B page, thereby using one of the 16 million record numbers available. In V7R1.0 you can store the extension records in Table X, freeing the 16 million record numbers in Table B for base records only. This allows storage of the maximum possible number of logical records in a file regardless of the number of extension records.

The record limit for Table B is BSIZE * BRECPPG, which can have a maximum value of 16777216. If the BSIZE * BRECPPG product exceeds that value, the CREATE command is rejected. Table X has the capacity to hold 512 million extension records, 32 times the Table B record limit. When the capacity of Table B or Table X is exceeded, the file is marked full.

Limit the size of the base records

File parameters RECRDOPT (record options) and BRLIMSZ (limit the size of base records) are used to limit the size of a base record and allow to place BRECPPG records on each Table B page. These parameters can be set only at file create time and are not resettable. To implement these capabilities for existing files, they must be reorganized under V7R1.0.

Using RECRDOPT=1

When RECRDOPT=1, the size of a base record is limited so all record numbers (also called record slots) on a Table B page can be utilized to store the base record of a logical record. RECRDOPT =1 can only be set if XSIZE is set to value greater than zero.

Files with RECRDOPT=1 are able to achieve the maximum capacity of (BRECPPG*BSIZE) logical records in entry order and unordered files. This means that a file may have 16,777,216 logical records regardless of the presence of extension records. Extension records do not reduce the number of logical records that may be stored, as is the case in files with XSIZE=0.

When XSIZE is greater than zero and RECRDOPT=0, the base record may grow almost up to a page size and only when space on the Table B page is exhausted does Model 204 start to use Table X for extensions. No extensions are allocated in Table B when Table X is defined regardless of RECRDOPT setting.

BRLIMSZ - base record limit size

The BRLIMSZ parameter is view-only. The value is calculated at file create time based on the value set for BRECPPG at file create time.

When RECRDOPT=1, the value of BRLIMSZ is calculated as follows:

- Entry order file

  ```
  BRLIMSZ=(6140/BRECPPG)-6
  ```

- Unordered file

  ```
  BRLIMSZ=(6136/BRECPPG)-6
  ```

Impact of setting RECRDOPT=1

In addition to maximizing logical record storage in Table B, the speed of storage and update of Table B records is increased. This results from eliminating the constraints processing for Table B page space. Constraint processing for Table B record slots remains active. This is also the case for files with BRECPPG=1 or XRECPPG=1.

- Using RECRDOPT=1 may result in not using all available space on a Table B page.
- If RECRDOPT=1, the maximum allowed value of BRECPPG is 1022 for ordered files and 1023 for unordered files.

Preallocated fields are limited by the base record size. BRECPPG for files with RECRDOPT=1 and preallocated fields must be chosen so that all preallocated fields fit into the base record size. Furthermore, the combined length of all preallocated fields cannot exceed BRLIMSZ.

Using Table X for extension record storage

If you create a file with RECRDOPT=1, then you must set XSIZE to a value greater than zero to establish Table X for storing extension records, because only base records will be stored in Table B. To implement this record storage methodology for existing files, they must be reorganized in V7R1.0.

- Files created with XSIZE greater than zero will have a FICREATE=7 setting and cannot be opened in prior releases.
- XSIZE greater than zero can be specified only if the file organization is Entry Order or Reuse Record Number.

In addition to XSIZE, the following new parameters apply to Table X: XAUTOINC, XHIGHPG, XQLEN, XRECPPG, XRESERVE, XREUSE, and XREUSED. For detailed documentation of each, see *Chapter 6: New and Enhanced Parameters and Commands.* The VIEW TABLES command will display these parameters only if the file has a Table X.

Command support for Table X

TABLEX command

The TABLEX command displays the space utilization of Table X.

```
TABLEX
479     AVG. FREE SPACE PER PAGE
18      AVG. FREE SLOTS PER PAGE
13      NUMBER OF PAGES PROCESSED


100     XRECPPG - TABLE X EXTENSION SLOTS PER PAGE
17      XRESERVE - TABLE X RESERVE SPACE PER PAGE
```

TABLEBX command

The TABLEBX command displays the space utilization for both Table B and Table X.

```
TABLEBX
4556    AVG. FREE SPACE PER PAGE
234     AVG. FREE SLOTS PER PAGE
2       NUMBER OF PAGES PROCESSED

500     BRECPPG - TABLE B RECORDS PER PAGE
17      BRESERVE - TABLE B RESERVED SPACE PER PAGE
479     AVG. FREE SPACE PER PAGE
18      AVG. FREE SLOTS PER PAGE
13      NUMBER OF PAGES PROCESSED

100     XRECPPG  - TABLE X EXTENSION SLOTS PERPAGE
17      XRESERVE - TABLE X RESERVE SPACE PER PAGE
```

Additional details regarding Tables B and X can be obtained with other options of the commands such as:

```
TABLEX LIST

TABLEBX LIST

TABLEX PAGES n TO n

TABLEBX PAGES n TO n
```

TABLEn RECLEN command for Table X

The RECLEN output changed in V7R1.0 to support Table X.

If XSIZE=0

- TABLEB RECLEN command generates the average *logical* record length, meaning a base record and its extension records, all in Table B.
- TABLEX RECLEN command generates the following message:

  ```
  ***M204.2768: FILE filename HAS NO TABLEX PAGES
  ```

- TABLEBX RECLEN also generates the average *logical* record length.

If XSIZE>0

TABLEB RECLEN command generates the average:

- Base record length in Table B. No Table X extension record information is included.
- Individual extension record length in Table X.
- Logical records length, meaning the base record in Table B and the extension records in Table X.

Commands updated to support Table X

The following commands have been modified to accommodate Table X.

COMPACTB	CREATE FILE	DECREASE
DUMP FILE	FILELOAD	FLOD
INCREASE	INITIALIZE	OPEN FILE
RENAME FILE	RESTORE FILE	TABLEB

Creating a file with Table X

In the following example, when XSIZE is set greater than zero, Table X is established for the VEHICLES file.

```
CREATE FILE VEHICLES
   PARAMETER FILEORG=X'24'  */Unordered, RRN file organization      /*
   PARAMETER BSIZE=128
   PARAMETER BRESERVE=100   */100 free bytes are required to store  /*
                            */a new record on page                  /*
   PARAMETER BREUSE=30      */when 30% or more page space is free,  /*
                            */put page on reuse queue               /*
   PARAMETER XSIZE=600
   PARAMETER XRESERVE=800   */800 free bytes are required to store  /*
                            */a new record for Table X on page      /*
   PARAMETER XREUSE=15      */when 15% or more page space is free,  /*
                            */put page on reuse queue               /*
END
```

Sizing and using Tables B and X

Preallocated fields

Preallocated fields may reside only in Table B records. Model 204 will never store them in Table X.

Model 204 will store non-preallocated fields in Table B records. However, when a given Table B record has no more room for additional non-preallocated fields, those fields will be stored in Table X extension records. The fields stored in Table X records have exactly the same format and space requirements as fields stored in Table B records.

Table B overhead

For files without a Table X, each Table B record has five bytes of overhead made up of a 3-byte extension pointer and a 2-byte slot number. For files with XSIZE greater than 0, each Table B record has six bytes of overhead made up of a 4-byte extension number and a 2-byte slot number.

- When XSIZE is set to 0, Table B must be sized such that it can contain all visible fields in all records.
- When XSIZE is greater than 0, the total size of Table B and Table X must be such that each visible field in all records will be stored in Table B or Table X.

There are many possible combinations of BSIZE and XSIZE that meet this requirement. So, for a file with a Table X, there is no formula for determining a unique BSIZE or XSIZE.

Table X overhead

The purpose of Table X is to free record slots in Table B that might have been used for extension records. There may be a performance side effect with using Table X. By experimenting with different values of XRECPPG, it may be possible to reduce the size of record extension chains—that is, have fewer but larger extension records instead of many smaller extension records. This would potentially reduce I/O required to read in very large records, such as those with many extensions.

Sizing tables with XSIZE greater than zero

Setting a default for XSIZE depends on the difference in the size of your records. The more variation in the length of your records, the more likely that you will have extension records and, therefore, need more Table X pages. CCA recommends the following: if the size of your records varies by 10%, then allocate 10% of the pages in Table B for Table X.

If XSIZE is greater than 0, the following formula can be used to size Table B:

```
BSIZE=1.2 *(total number of base records) / BRECPPG
```

And the following formula can be used to size Table X:

```
XSIZE=1.2 *(total number of extension records) / XRECPPG
```

Note: Table X slots are always reused after extension records are deleted. Table B slots are reused only for Reuse Record Number (RRN) files.

COMPACTB data compaction enhancements

The COMPACTB process has been enhanced to provide better compacting results and improved performance. In addition, the following features have been added to COMPACTB:

- Support for files with Table X
- A DELETE option to physically delete logically deleted records

  See “COMPACTB: Invoking the data compactor” on page 6-31 for the command syntax.

COMPACTB processing can be used to avoid frequent file reorganizations needed to reduce the number of extension records and can also be used to reclaim space occupied by logically deleted records—some times referred to as dirty data.

Recovery of the compaction process is fully supported.

Improved compaction for all files

If a Table B base record has one and only one extension record and the base record page has sufficient free space, then the data compactor will combine the extension record with the base record and delete the extension. For a base record with multiple extensions, the compactor will try to combine the extension records into fewer extensions, but will not attempt to combine them with the base record.

Compaction for files with Table X

COMPACTB processing also supports files defined with Table X. Compaction for files defined with Table X will have better performance, because compaction now operates on a page basis instead of a record basis and locks fewer resources, especially the existence bit map.

Physical delete of logically deleted records

A logically deleted record remains physically stored in the file. The corresponding existence bit is zero (or off) in the Existence Bit Pattern (EBP). But otherwise, all fields, data, and index entries are present and consume space. However, you cannot access the record, because the existence bit is zero.

The data compactor can physically delete logically deleted records in files, if Table X is defined for the file. The compactor accomplishes this by checking all records on a Table B page to verify that the EBP indicates that the record exists. For records that are physically present, but the EBP indicates that the record has been logically deleted, the compactor will delete the physical records—if requested.

The new DELETE option of the COMPACTB command specifies that the process must physically delete logically deleted records. A deletions can occur whether the physical record is in only Table B or the base record is in Table B with extension(s) in Table X.

Space and record numbers freed by physically deleting logically deleted records are reusable.

You can use the DELETE option with all other options. For example:

```
COMPACTB FROM 15 TO 23 FREE 20 DELETE
```

The data compactor finds logical records with a nonzero length within the indicated record number range (in the previous example, 15 to 23) that do not exist in the EBP and deletes the data and index entries for such records. This example also releases 20% of the unused or free pages to the data compactor, which can be used for new extension records. Files with lock pending updates (LPU) disabled are processed with the data compactor placing an exclusive enqueue on the entire file. In this case, a warning message is issued.

For files with many logically deleted records, processing the DELETE option may be CPU and I/O intensive. The following new message is printed at the completion of data compactor processing that states the total number of deleted records.

```
M204.2754: NUMBER OF DELETED LOGICALLY DELETED RECORDS:
count
```

All statistical information from compaction is presented as Model 204 messages, as shown in the following COMPACTB command outputs.

COMPACTB using the FREE option

```
COMPACTB FREE 85
*** M204.2749: NUMBER OF BASIC RECORDS PROCESSED:                36000
*** M204.2750: NUMBER OF EXTENSION RECORDS BEFORE COMPACTION:    110001
*** M204.2751: NUMBER OF EXTENSION RECORDS AFTER COMPACTION:     35991
*** M204.2752: NUMBER OF NOT PROCESSED (LOCKED) RECORDS:         0
*** M204.2753: NUMBER OF FREE PAGES USED:                        1026
```

COMPACTB using the DELETE option

```
COMPACTB DELETE

***  M204.2749: NUMBER OF BASIC RECORDS PROCESSED:               35000
***  M204.2750: NUMBER OF EXTENSION RECORDS BEFORE COMPACTION:   106952
***  M204.2751: NUMBER OF EXTENSION RECORDS AFTER COMPACTION:    34991
***  M204.2752: NUMBER OF NOT PROCESSED (LOCKED) RECORDS:        0
***  M204.2753: NUMBER OF FREE PAGES USED:                       755
***  M204.2754: NUMBER OF DELETED LOGICALLY DELETED RECORDS:     1000
```

Automatic increase for Tables B, D, and X

The automatic increase option for Tables B, D, and X lets table increases occur without manual intervention. If a file has been created with this capability, an automatic increase will occur when the table becomes full due to lack of allocated pages in Tables B, D, and/or X.

Large transaction volumes or significant data growth can increase page requirements beyond what is currently allocated to the file tables. When this occurs, files are marked full, requests are cancelled, and transactions are backed out. Consequently, files may be left unusable for update processing and applications until a manual intervention increases the table size.

Implementation of automatic increase capabilities in your files can reduce request cancellations and transaction back outs due to table full conditions. This feature expands the ability to provide 24/7 file and application availability.

The page increase is based on the values of the controlling parameters that are set by the file manager or system defaults that can be set by a system manager.

- For files created prior to V6R1.0 the system parameters, if set, will be used to increase supported tables that fill.
- For files created at V6R1.0 or later the file and system parameters, if set, will be used for automatic increase operations.

If the file parameters are not set, but the system parameters are set, they will be used. If both file and system parameters are set, the file parameters take precedence. If none of the automatic increase parameters are set, no automatic increases can occur.

Automatic increase usage

The automatic increase operation for Tables B, D, and X can add up to FREESIZE pages to those tables when the file full condition occurs and to avoid the negative consequences of a table filling. Automatic increase will automatically reallocate pages from FREESIZE, if pages are available, to the appropriate table. The page increase is based on the values of the controlling parameters that are set by the file and system managers.

- When one of these tables fills, only that table is increased.
- A following message is sent to the audit trail regarding an automatic increase event. Also, this event creates a discontinuity and the information is written to the journal.

  ```
  M204.2835: %F TABLE %C INCREASED BY %C PAGES
  AUTOMATICALLY
  ```

- The automatic increase operation is a non-backoutable transaction, even if the file full event is backed out. In this case the increased table size will be preserved.
- Automatic increase for Tables B and D is supported for entry order, unordered, and sorted files. Entry order and unordered organizations also support the automatic increase of Table X. For hash key files automatic increase is supported only for Table D.

Enabling automatic increase

To use automatic increase the files must be created under V6R1.0 or later. The file and system parameters that control automatic increase are: BAUTOINC, DAUTOINC, XAUTOINC, DEFINCP, and MAXINCBP. See Chapter 6: *New and Enhanced Parameters and Commands* for the details on setting and resetting these parameters.

To implement automatic increase capabilities for files created prior to V6R1.0 you must reorganize the files under a supporting Model 204 release.

Table B automatic increase - BAUTOINC and/or DEFINCP and MAXINCBP

Set the BAUTOINC file parameter to the number of pages you want reallocated to Table B from FREESIZE pages when a Table B full condition occurs. If BAUTOINC is set to 0, the value of the DEFINCP system parameter is checked. If DEFINCP is a non-zero value, Table B is increased by DEFINCP percentage of BSIZE pages up to a limit set by the MAXINCBP parameter.

DEFINCP parameter details

DEFINCP can be set in the job stream or set by a system manager.

The DEFINCP value represents the percentage of the current table size by which that table can be automatically increased if it fills during processing. DEFINCP applies to Tables B, D, and X based on file organization.

Understanding the MAXINCBP parameter

The MAXINCBP system parameter sets a limit, as a percentage of BSIZE, on how much Table B can increase by. This percentage is then used to calculate the maximum number of segments, rounded up to a whole number that the file could have if Table B were increased by the entire amount. The actual number of pages that BSIZE can increase to is determined by the calculated number of segments, divided by BRECPPG.

The compiler uses the maximum possible number of segments to determine the correct length of the entries it allocates in the resource locking table and in VTBL. During compilation the file is treated as though it has already been increased to the maximum number of segments (MAXINCBP) even though no increase may yet have occurred.

The total Table B expansion with no MAXINCBP is limited to existing number of segments. MAXINCBP defines the maximum size of Table B, but if DEFINCP is not set nor the file's BAUTOINC, then auto increase does not happen. If a file is opened without MAXINCBP set yet, then no increase in number of segments is possible. MAXINCBP must be set (or reset) before the file is opened.

Example of Table B automatic increase

Take a file with BSIZE=100000, BRECPPG=10. For purposes of allocating entries in the record locking table, the compiler treats this file as though all records are present and it is therefore, a 21-segment ((10*100000)/49152) file. However, if the system parameter MAXINCBP=20, then Table B in this file could increase by 20,000 pages. The resulting Table B of 120,000 pages would be 24.4-segment ((120000*10)/49152) segment file. This would be rounded up to 25 segments, so BSIZE could actually be increased to 122,880 before crossing into the twenty-sixth segment.

These calculations occur when each file is opened for the first time and affect the compilation of requests from that time forward, for that run.

Increasing LRETBL to accommodate MAXINCBP

Setting MAXINCBP might require a corresponding increase to LRETBL.

Table D automatic increase - DAUTOINC and/or DEFINCP

Set the DAUTOINC file parameter to the number of pages you want reallocated to Table D from FREESIZE pages when a Table D full condition occurs. If DAUTOINC is found to be set to 0, the value of the DEFINCP system parameter is checked and Table D is increased by DEFINCP percentage of DSIZE.

DEFINCP can be set in the job stream or set by a system manager. The DEFINCP value represents the percentage of the current table size by which that table can be automatically increased if it fills during processing. DEFINCP applies to all files for Tables B, D, and X.

Table X automatic increase - XAUTOINC and/or DEFINCP

Set the XAUTOINC file parameter to the number of pages you want reallocated to Table X from FREESIZE pages when a Table X full condition occurs. If XAUTOINC is found to be set to 0, the value of the DEFINCP system parameter is checked and Table X is increased by DEFINCP percentage of XSIZE.

You can implement the XAUTOINC parameter for files created in V7R1.0, or later, that have a Table X, and you can view it. A file has a Table X when XSIZE is greater than zero.

- If there are enough FREESIZE pages, the auto increase takes place.

- If there are some FREESIZE pages, but not enough to satisfy the XAUTOINC value, the auto increase proceeds until FREESIZE is exhausted.
- More pages may be added to Table X to add one more segment to the file. See the *Model 204 File Manager's Guide*.

When the file becomes full from lack of space in Table X, an automatic increase is done by comparing and using the smaller number of the following:

- The value of XAUTOINC
- The number of pages in the free space

DEFINCP can be set in the job stream or set by a system manager. The DEFINCP value represents the percentage of the current table size by which the table can be automatically increased if it fills during processing. The DEFINCP value applies to all files and for Tables B, D, and X based on file organization.

When an automatic increase takes place

If FREESIZE is 0 or all automatic increase parameter values are 0, no automatic increase can occur and the default processing for a table full condition results.

When a file is opened, no automatic increase occurs for any table. This avoids increasing a file at open time that may have problems with recovery. The DEFINCP parameter can be set with a default percentage increase for Tables B, D, and X and the table lacking space will be increased dynamically. When a table becomes full, Model 204 checks the corresponding automatic increase value--BAUTOINC, DAUTOINC, or XAUTOINC. If it is zero, then it checks DEFINCP. If it is not zero, the table is increased by DEFINCP percent.

Reclaiming table space

XREUSE

If XREUSE is set fairly high and Table X fills up to the point where the reuse queue is empty and there is no room to increase XHIGHPG, considerable space might still be available on most pages. To prevent a file full condition, Model 204 tries to find space to add a new record on as many as 16 randomly selected pages. Under these circumstances, issue an INCREASE command for Table X or set XREUSE lower.

The Table X reuse queue is built for only unordered or entry order files for which XSIZE is greater than zero.

COMPACTE data compaction for Table E

Table E compactor and TABLEE command

Model 204 stores large objects as consecutive chunks of Table E pages. When large objects are created and deleted frequently, gaps can occur between objects that may not be reused due to their small size. The COMPACTE command lets you compact Table E by grouping gaps together, thus reducing Table E fragmentation. To find usable gaps that may be compacted, the Table E map must be analyzed.

The Table E compactor can combine orphan spaces in Table E without file reorganization and run without exclusive use of file. When processing finds a gap, the large object that follows the gap is switched with the gap. The large object moves left, concentrating objects at the beginning of Table E, while the gap moves right, concentrating free space at the end of Table E. Although a Large Object may be pointed to by one and only one record, different fields in the same record may point to different Large Objects.

Introducing the Large Object header

To make the Table E compaction process work each large object starts with a header that is new in V7R1.0. (The object descriptor in the Table B record has not changed and the large object header length is not included in the large object descriptor length.) The large object header contains a field for the Table B record number that points to the large object—thus a backward pointer to the Table B record.

Implementing a large object header requires file reorganization. Creating the file anew establishes a new file version. Only files with the new version are eligible for COMPACTE processing. No application changes are required for the new file version.

Note: Files created in V7R1.0 with Table E size greater than zero are not backward compatible and cannot be opened in earlier releases.

When each large object is stored, the new header is also included. The large object header requires the following additional storage and must be calculated for Table E sizing requirements.

The large object header has the following 4-byte fields:

- Table B record number
- Large object length in pages, including reserved pages
- Field attribute

  The field attribute facilitates the Table B record search to find a field with the object descriptor. The header length is 32 bytes.

Considerations for compacting Table E

Some compactions may be counter productive. For example, if a segment has 49 objects, each the size of 1000 pages, and 49 gaps of three pages each for a total size of 149 pages, then moving 49,000 pages to reclaim a modest 147 page gap is inefficient. On the other hand for objects with average size of 1-100 pages, compacting a hundred 1-page gaps is beneficial.

The TABLEE command, like the TABLEB command, reports Table E usage statistics: the number of gaps and total gap size. Because compaction is heavily I/O and CPU intensive, you should compact Table E only when you can expect substantial results.

For files with large Table E and really large objects (thousands of pages) you must take care to prevent unnecessary page movements.

The compactor analyzes Table E on a segment by segment basis, where each segment represents 49,152 pages of Table E.

Table E contains not only object pages but bitmap pages also. The current compactor's implementation has the following limitations:

- Bitmap pages allocated one per segment are not moved, so the worst result of compaction is two gaps per segment.
- Objects residing in more than one segment are not moved.

Using the TABLEE and COMPACTE commands

To effectively compact Table E, CCA recommends running a TABLEE command with the SEG option, identifying segments with large number of gaps, running COMPACTE command for segments of interest, and then running another TABLEE command for compacted segments to check the results.

COMPACTE back out and recovery

No back out capabilities are provided for Table E compaction.

To facilitate recovery, the compactor writes preimages of all large object pages that are subject to move. You may need to increase checkpoint dataset size. In the worst case almost all pages in Table E may be preimaged.

The journal dataset size increase is much smaller. It writes 50 bytes per object moved. If a problem happens during compaction, base the recovery action on error messages.

- For error messages generated while analyzing Table E (messages 2809, 2810, 2818, 2819, 2821), a file must be regenerated.
- For error messages generated while moving an object (messages 2811, 2823) a normal file recovery should be adequate.

If the problem persists, you must regenerate the file.

COMPACTE performance

Table E compactor processing is highly I/O and CPU intensive. When gaps combine and grow in size, it may be quite expensive to do page-by-page constraints checking. Use of EXCL option lets you avoid constraints checking, but the total file will be unavailable to other users for the duration of compaction.

COMPACTE and checkpoint

The COMPACTE command runs as one long transaction. After reading the MAXPR (number of pages), processing stops, the transaction ends, and a checkpoint is attempted. Also, at this time processing checks whether the user is being bumped or is exceeding limits, such as I/O or CPU slices or a higher priority user needs to run. These checks happen only after an object has been moved. If a very long—hundreds of pages—object is moved, the transaction or sub transaction checkpoint may be delayed or prevented.

3

Date/Time Stamp Support

In this chapter

- Introduction
- Enabling date/time stamp updates
- Suspending date/time stamp updates
- Extracting date/time stamp field data
- Using the date/time stamp feature

Introduction

The date/time stamp feature lets you put a date/time stamp (Limited) on each record that was updated in a transaction. You can then use the data in the date/time stamp field in an end user written application to find and process all the rows of data that were changed.

An application that processes these records can track the date/time stamp field values that have been processed to date, or it may take another action, including updating the date/time stamp field.

To install and use this functionality, you must obtain the proper decrypt keys and install this release.

Deleting records

The date/time stamp feature does not include support for DELETE RECORD or DELETE RECORDS processing. DELETE RECORD or DELETE RECORDS processing must be handled by your application software.

As well, you can use logical delete techniques. However, in all forms of deleting records, it is your responsibility to maintain a log of record deletions, if you want one.

Managing date/time stamp performance

Plan to have your hardware, operating systems and Model 204 partitions, servers, files and parameters configured to support the additional overhead required. Putting a date/time stamp on records requires the same system resources as any Model 204 field (and index, if the field is indexed) update, including the possibility of the following events:

- File Full messages for Tables A, B and D, as well as C, if using a HASH key for the date/time stamp field, which implies a possible file reorganization.
- Adding extension records.
- Resetting the BRECPPG, BRESERVE and other INDEX-related parameters for participating files.
- Splitting an index.
- Flushing the journal buffer.
- Increased CCAJRNL dataset requirements.
- Increased CHKPOINT dataset requirements.
- Increased Data Base File Size requirements.
- Filling CCATEMP.
- Increased CCATEMP requirements for the:

  - Transaction back out log
  - Constraints database
- Increased CPU, DKPR, DKRD, DKWR, and RQTM, and similar statistic values.
- Server size increase for applications supporting date/time stamp files.
- Increased server swapping.
- Longer time required for update transactions.

Handling replication

Should you wish to use the DTS feature to enable bidirectional replication, please note the significant omissions in the bidirectional replication arena that date/time stamp feature does not support—specifically:

- Coordinated recovery
- DELETE RECORD and DELETE RECORDS commands

  No date/time stamp field update is recorded when another field is physically deleted from the database by a file manager.

  As you delete a field from an individual record, the date/time stamp value is updated.
- FILE LOAD and FLOD commands
- REDEFINE FIELD and RENAME FIELD commands

  The value of the date/time stamp field is not updated when you redefine a field or rename it.
- RESTORE processing
- REGENERATE processing

Compatibility

A file must have been created in V6R1.0 or later to use the date/time stamp feature. An attempt to use the date/time stamp feature with files created by an earlier release generates an error:

```
M204.2733: DTS OPTION IS NOT ALLOWED FOR FILES CREATED
BEFORE RELEASE 6.1
```

Enabling date/time stamp updates

To activate date/time stamp updates, see “Installing the date/time stamp feature” on page 1-34. Then, take the following steps.

1. If you did not specify the date/time stamp field name by modifying the CUST source code for installation, you can specify the name now by setting the User 0 CCAIN parameter DTSFN.

   Model 204 first checks the DTSFN parameter value, then the CUST module. If you have not specified a value in either place, it is not possible to open an existing date/time stamp file or create a new one. Attempting to do so generates the following error:

   ```
   M204.2729: DATE/TIME STAMP FILE, BUT DTSFN UNSPECIFIED
   FOR RUN
   ```

   At CREATE time, the file is not a date/time stamp file. It is not a date/time stamp file until the FOPT=X’10’ has been set.

2. Define the date/time stamp field in each file that you want to have date/time stamp updates.

   You must open the file with file manager privileges to define the date/time stamp field. See “Specifying a date/time stamp field definition” on page 3-6 for details.

   Remember, there is no user/logon privilege that represents file manager. File manager privileges are file-level privileges established on the privileges defined to the password used to open the file or the PRIVDEF (default) file privileges when no password is supplied on the file open.

3. Set FOPT to include X’10’ in each date/time stamp file.

   Do this *after* the date/time stamp field is defined in the file. An FOPT value including X’10’ cannot be set on file create. If you attempt to reset the FOPT bit before defining the date/time stamp field, then the following error is issued:

   ```
   M204.2730: DATE/TIME STAMP FILE, BUT DTSFN NOT DEFINED
   IN FILE
   ```

Limitations on a date/time stamp enabled file

The Date/Time Stamp feature does not support:

- Parallel Query Option. If you attempt to use the feature while using PQO/204, you invoke the following message:

  ```
  M204.1977: %F MAY NOT BE ACCESSED REMOTELY
  ```

- IFAM. If you attempt to use the feature while using IFAM, you invoke the following message:

```
M204.2831: IFAM DOES NOT SUPPORT DTS FILES
```

- SQL processing

Furthermore, once the FOPT X’10’ bit has been turned on, the file can be opened only under certain conditions:

- The date/time stamp field name must be specified via the installed CUST code or the DTSFN parameter.
- Under Model 204 V6R1.0, the file can be opened, but the date/time stamp field will not be updated.

Note: Model 204 releases earlier than V6R1.0 cannot open a file created by V6R1.0 or later

Date/time stamp file behavior

A file is identified as being a date/time stamp file by the FOPT value X’10’. When this option is set, any records that are updated in the transaction also have their date/time stamp fields updated as the first step in COMMIT processing.

A file with the FOPT value X’10’ must be a Lock Pending Updates (LPU) file. An attempt to set FOPT to include both X’10’ and X’02’ for a file results in the error:

```
M204.2726: FILE OPTIONS X’10’ AND X’02’ ARE INCOMPATIBLE
```

Date/time stamp field updates are done at COMMIT time, either implicit or explicit. The value in the date/time stamp field is the time that commit processing begins, not the time of the actual commit, which follows the DTS updates. The date/time stamp value is recorded after all updates for the transaction have completed.

The X’10’ FOPT value may not be specified as a CREATE parameter, since no date/time stamp fields are defined at that time. Attempting to do so results in the message.

```
M204.2730: DATE/TIME STAMP FILE, BUT DTSFN NOT DEFINED IN
FILE
```

If you initialize a date/time stamp file and do not specify the KEEPDEFS option, the file will no longer have a date/time stamp field. In this case, date/time stamp processing is disabled and a message issued:

```
M204.2736: FILE DTSFILE FOPT=X'10' TURNED OFF: DATE/TIME
STAMP FIELD UNDEFINED
```

Date/time stamp field behavior

Specifying the date/time stamp field name

The new system parameter, DTSFN, specifies the date/time stamp field name. This field name is the same for all files participating in DTS processing in a run. Though this is a system-level parameter, the feature is a file-level feature.

The date/time stamp field name you specify is not edited for validity.

Although the date/time stamp field name is the same for all date/time stamp files in a particular run, as specified by DTSFN, it is not necessarily always the same for a given file. You can define several date/time stamp fields in the same file. However, the active DTS field will be the one that matches the name specified by DTSFN *in that run.*

The DTSFN value may be specified in CUST source code, so you do not have to change all your jobs to specify the parameter. See “DTSFN: Name of the date/time stamp field” on page 6-9. As shipped, Model 204 does not provide a field name.

Specifying a date/time stamp field definition

A date/time stamp field is defined as is any other Model 204 field. The restrictions on permissible field attributes are:

- May not be INVISIBLE or UNIQUE.
- If preallocated, OCCURS may not be greater than 1.

Disregarding these restrictions causes the following error:

```
M204.2728: DATE/TIME STAMP FIELD MAY NOT BE INVISIBLE,
UNIQUE, NOR HAVE MULTIPLE OCCURRENCES
```

CCA recommends using the ORDERED CHARACTER attribute. By default, Model 204 generates a character string date/time stamp field value.

Defining the date/time stamp field value format

Your custom code formats the data entered into the field. The format for the default date/time stamp string is:

```
CCYYMMDDHHMMSSMMM
```

Note: The DTSFN field may have differing definitions in different Model 204 databases.

Suspending date/time stamp updates

A user may exclude date/time stamp files from DTS processing. Turning off DTS updates lets you update any field, including the date/time stamp field without automatically updating the date/time stamp field.

To suspend DTS updates for a user's thread, reset DTS to 0. To do so, the user must have system manager privileges. Any open date/time stamp files must have been opened with file manager privileges. Any additional date/time stamp files the user opens after the reset must also be opened with file manager privileges.

To disable date/time stamp processing for a specific file, you can turn off the FOPT X'10' bit. However, doing so disables date/time stamp updates made by all users of that file.

Note: Automatic date/time stamp field updates are still performed for updates made to the file(s) by other users using DTS=1.

Extracting date/time stamp field data

Since the date/time stamp field is a normal Model 204 field, all methods of extraction may be used.

In User Language

The date/time stamp field may be:

- Read and printed either as a field name reference or via a $function.
- Searched in normal FIND statements.
- Assigned to %variables or other fields.
- Printed as supported for all Model 204 fields, including a PAI statement.

OPEN processing

If the specified date/time stamp field name is not defined in a date/time stamp file, an OPEN command is rejected. If the date/time stamp field name is DATE.TIMESTAMP, for example, the following messages are issued:

```
> OPEN DTSFILE
***  1  M204.1265: NO SUCH FIELD NAME. FIELD = DATE.TIMESTAMP
***  M204.2730: DATE/TIME STAMP FILE, BUT DTSFN NOT DEFINED IN FILE
***  M204.0608: FILE CLOSED: DTSFILE
***  2  M204.0630: FILE OPEN COMMAND REJECTED
```

If you do not set the DTSFN system parameter, files with FOPT=X’10’ cannot be opened. An OPEN command will not be processed and the following message is issued:

```
M204.2729: DATE/TIME STAMP FILE, BUT DTSFN UNSPECIFIED IN
RUN
```

If you have reset DTS to 0, you must open DTS enabled files with file manager privileges. Otherwise, the open is rejected.

Note: Model 204 V6R1.0 does not prevent the opening of a file that has the FOPT=X’10’ parameter set with V7R1.0 or later. You can open a FOPT=X’10’ file with a V6R1.0 release and update data, however, the date/time stamp field will not be automatically updated.

Using backup and recovery

If any errors occur during processing, the transaction is backed out.

Backup and recovery activity synchronize the date/time stamp with the data changes made. Date/time stamp field updates are part of the customer’s

updating transaction and follow the normal Model 204 rules for BACKOUT, COMMIT, DUMP, RESTORE, ROLL BACK, and ROLL FORWARD.

Using the file load utilities

FLOD and FASTLOAD do not update the date/time stamp values. However, you can manually update date/time stamp values as you might update another field name value pair.

Using the date/time stamp feature

The field defined as the DTS field for a file can occur more than once on a record if the field is defined as non-preallocated. However, only the first occurrence on any record will be maintained by DTS with the date/time stamps.

Ensuring the date/time stamp field is updated within a procedure

If there are two requests in one procedure, for example, and neither COMMIT nor CLOSE is called between the two, then both requests are one transaction.

The following procedure has two requests.

```
BEGIN
   STORE a record in a DTS file
END
BEGIN
   PAI the record
END
```

DTS updates are not displayed by the PAI output, since the transaction does not complete until the end of the procedure.

- If there is no COMMIT in the first request or if the file is not closed after the first request, then the date/time stamp field is not updated.
- If the same User Language requests are executed as separate procedures, the date/time stamp field is updated.

Setting the DTSFN parameter

If you set the DTSFN parameter and then define or redefine a date/time stamp field, Model 204 will promptly verify invalid attributes.

If you have already defined a field as your date/time stamp field prior to setting the DTSFN parameter, the attributes for the date/time stamp field are not verified for validity until DTS processing begins.

Limitations to date/time support

The date/time stamp feature is:

- Supported only for ONLINE and BATCH204
- Not supported under PQO
- Not supported under IFAM

4

MQ/204 Enhancements

In this chapter

- Overview
- Message groups
- Searching for messages using the MATCH options
- BROWSE options
- Special handling options
- Examples of writing messages and browsing groups
- Command and statements
- DEFINE QUEUE: Identifying a WebSphere MQ queue
- MODIFY QUEUE statement
- MQGET statement
- MQPUT statement
- MQPUT1 statement
- OPEN QUEUE statement
- Return codes and messages

Overview

Grouping messages

MQ/204 takes advantage of the new features IBM introduced for WebSphere MQSeries, in particular the grouping of messages. The new fields in the MQMD Version 2 are Group ID and status, sequence number and flags. The Group ID, sequence number, and some of the flags determine whether a message is part of a group. Now Model 204 users can control the grouping of messages.

The other fields that apply to segmentation are not covered, because the segmentation options are not available for z/OS.

Supporting Java Message Service (JMS)

MQ/204 supports the predefined JMS protocol. MQ/204 now understands and can talk to Java through the messaging services. The JMS protocol, when sending MQ messages, expects to encode certain Java-specific fields into the additional header known as the MQRFH Version 2 (MQRFH2) header. The receiving mainframe application should expect to receive such a header, if it knows that the source of the messages is a JMS application.

Because the MQRFH2 header carries JMS-specific information, always include it in the message when you know that the receiving destination is a JMS application.

Conversely, omit the MQRFH2 when sending a message directly to a non-JMS application, because such an application does not expect an MQRFH2 in its WebSphere MQ message.

Environment requirements

Minimum is Model 204 V7R1.0.

Consulting IBM documentation

CCA recommends that you keep handy:

IBM Application Programming Reference.

IBM MQ manual, *Using Java*

Message groups

You can group messages together. Each message in a group has its own sequence number and shares the group ID number. The sequence number is assigned by system or by you, depending on whether you select the LOGICAL_ORDER or NOT_LOGICAL_ORDER option on your command or statement.

- The physical order of the messages in the group is the order in which the messages were written and that order is retained using the LOGICAL_ORDER option, so that the physical and logical order of the messages in the group match.
- You can set the logical order of the messages in the group using the NOT_LOGICAL_ORDER and the SEQUENCE option. The physical and logical order of the messages in the group may differ, as you assigned the sequence numbers.

In either case, the messages in the group are processed in ascending order and the highest sequence number in the group is called the last logical message.

When you use MQPUT to write messages, you have three options, which are discussed in turn:

- Group messages in logical order
- Group messages, but not in logical order
- Do not group messages

Messages grouped in logical order

Grouping messages in logical order is the more simple way of writing messages that are in a group.

1. On the first message, you set the options:

   ```
   GSTATUS='G'

   LOGICAL_ORDER
   ```

   There is no need to set GROUPID or SEQUENCE, as the queue manager automatically assigns these values.

2. Continue writing messages to the queue with GSTATUS='G' and LOGICAL_ORDER options set until the last logical message

3. For the last message, you set the options:

   ```
   GSTATUS='L'.

   LOGICAL_ORDER
   ```

Using this method, you can write messages to only one group at a time.

Message groups and MQPUT1

The LOGICAL_ORDER option is not allowed on the MQPUT1 statement, although the GSTATUS, SEQUENCE, and GROUPID options are allowed, so it is possible to write a message to a group using MQPUT1, but not in logical order.

Messages grouped, not in logical order

If you need to group messages, but cannot use the LOGICAL_ORDER option because either you need to write

- To two or more groups at the same time
- Messages where the physical sequence is not the same as the logical sequence (You are not writing the messages in ascending order of sequence number.)

Grouping messages, but not in logical order, is more complex to handle, but gives you more flexibility.

1. On the first message, you must set the options:
   - GSTATUS to 'G', unless this is the last logical message in the group—the highest sequence number—in which case, set GSTATUS to 'L'.
   - NOT_LOGICAL_ORDER
   - GROUPID to null or spaces
   - SEQUENCE to the sequence number of this message within the group, which must be greater than zero, although not necessarily one, because the messages can be out of sequence.
2. When this first MQPUT to the queue completes, you must save the GROUPID returned in the DESCRIPTOR field, because this is the Group ID that the queue manager has assigned to this group. You will set this for all subsequent messages in this group.
3. For all subsequent messages in the group, set the options:
   - GSTATUS='G', unless this is the last logical message in the group—the highest sequence number—in which case set GSTATUS='L'.
   - GROUPID to the value returned from the first MQPUT (step 2)
   - NOT_LOGICAL_ORDER
   - SEQUENCE to the sequence of this message within the group.

If you write groups in this way, you can write to several groups at the same time. Be careful to save all the Group ID values from the first MQPUT for each group.

Messages not grouped

Messages not grouped was the only available behavior in MQ/204 V6R1.0 and earlier. You can maintain previous behavior by omitting GSTATUS or setting it to null or space (' '). The settings for SEQUENCE, GROUPID and LOGICAL_ORDER options are ignored.

Searching for messages using the MATCH options

You can retrieve messages that match criteria that you specify. The new options for the MQGET statement are:

```
MATCH_CORREL_ID or NOT_MATCH_CORREL_ID

MATCH_GROUP_ID

MATCH_MSG_ID or NOT_MATCH_MSG_ID

MATCH_MSG_SEQ_NUMBER

MATCH_MSG_TOKEN
```

Although these options are new, the behavior prior to MQ/204 in V6R1.0 and earlier was to match on any supplied CORRELID and MSGID. To avoid upward compatibility issues, the MQ/204 V7R1.0 defaults are:

```
MATCH_CORREL_ID

MATCH_MSG_ID
```

There are no equivalent NOT_ options for the criteria of GROUPID, SEQUENCE, and MSGTOKEN, because simply omitting these MATCH_ options reverses their effect.

Before searching for and retrieving messages

To search and retrieve messages using the MATCH options, you must know the index type of the queue. The index type of the queue was set by the system manager when the queue was defined.

Table 4-1 lists the index types, their purpose, and usage.

Table 4-1. Index types and purpose

Index type	Queue manager maintains	For queues that…
CORRELID	Correlation identifiers of the messages on the queue.	The application usually retrieves messages using the correlation identifier as the selection criterion on the MQGET call.
GROUPID	Group identifiers of the messages on the queue.	The application retrieves messages using the LOGICAL_ORDER option on the MQGET call.
MSGID	Message identifiers of the messages on the queue.	The application usually retrieves messages using the message identifier as the selection criterion on the MQGET call.

Table 4-1. Index types and purpose (continued)

Index type	Queue manager maintains	For queues that…
MSGTOKEN	Message tokens of the messages on the queue for use with the workload manager (WLM) functions of z/OS.	Are WLM-managed queues; do not specify it for any other type of queue. Also, do not use this value for a queue where an application is not using the z/OS workload manager functions, but is retrieving messages using the message token as a selection criterion on the MQGET call.
No index	No index	Are usually processed sequentially, that is, without using any selection criteria on the MQGET call.

Using index types with message groups

Table 4-2 and Table 4-3 on page 4-8 list the search criteria you can use when a queue is defined with particular index type(s). Since you can retrieve messages in logical order (Table 4-3 on page 4-8) or not logical order (Table 4-2), your MQGET statement can also include either the NOT_LOGICAL_ORDER or LOGICAL_ORDER option.

Table 4-2. Index types and NOT_LOGICAL_ORDER option

Selection criteria on MQGET call	Index type for nonshared queue	Index type for shared queue
None	Any	Any
Selection using one MATCH_ option		
MSGID	MSGID recommended	None or MSGID
CORRELID	CORRELID recommended	CORRELID required
GROUPID	GROUPID recommended	GROUPID required
Selection using two MATCH_ options		
MSGID + CORRELID	MSGID or CORRELID recommended	MSGID or CORRELID required
MSGID + GROUPID	MSGID or GROUPID recommended	Not supported
CORRELID + GROUPID	CORRELID or GROUPID recommended	Not supported
Selection using three MATCH_ OPTIONS		
MSGID + CORRELID + GROUPID	MSGID or CORRELID or GROUPID recommended	Not supported

Table 4-2. Index types and NOT_LOGICAL_ORDER option (continued)

Selection criteria on MQGET call	Index type for nonshared queue	Index type for shared queue
Selections using group-related criteria		
GROUPID + SEQUENCE	GROUPID required	GROUPID required
SEQUENCE (must be at least one)	GROUPID required	GROUPID required
Selection using MSGTOKEN		
For application use	Do not index by MSGTOKEN	Do not index by MSGTOKEN
For Work Load Management (WLM) use	MSGTOKEN required	Not supported

Table 4-3 lists the required index type when LOGICAL_ORDER is specified on a shared or not-shared queue.

Table 4-3. Index type and LOGICAL_ORDER option

Selection criteria on MQGET call	Index type for nonshared queue	Index type for shared queue
None	GROUPID required	GROUPID required
Selection using one MATCH_ option		
MSGID	GROUPID required	Not supported
CORRELID	GROUPID required	Not supported
GROUPID	GROUPID required	GROUPID required
Selection using two MATCH_ options		
MSGID + CORRELID	GROUPID required	Not supported
MSGID + GROUPID	GROUPID required	Not supported
CORRELID + GROUPID	GROUPID required	Not supported
Selection using three MATCH_ options		
MSGID + CORRELID + GROUPID	GROUPID required	Not supported

Retrieving messages not grouped

If there is no current group or logical message, only messages that have SEQUENCE=1 are eligible for return. In this situation, you can use one or more of the following match options to select which of the eligible messages is the one actually returned:

- MATCH_CORREL_ID
- MATCH_GROUP_ID
- MATCH_MSG_ID

If LOGICAL_ORDER is specified and there is a current group, only the next message in the group is eligible for return; this cannot be altered by specifying MATCH_ options.

You can specify match options which are not applicable, but the value of the relevant field must match the value of the corresponding field in the message to be returned.

You can specify one or more of the following match options.

Table 4-4. Using the MATCH options

MATCH_ option	Retrieves message with specified	Message retrieved must have
CORREL_ID	Correlation identifier	Correlation identifier that matches the value of CORRELID, as well as any other matches that may apply, such as message identifier. If NOT_MATCH_ option is specified, the CORRELID field is ignored and any correlation identifier will match.
GROUP_ID	Group identifier	Group identifier that matches the value of GROUPID, as well as any other matches that may apply, such as correlation identifier.
MSG_ID	Message identifier	Message identifier that matches the value of MSGID, as well as any other matches that may apply, such as correlation identifier. If NOT_MATCH_ option is specified, MSGID is ignored and any message identifier will match.
MSG_SEQ_NUMBER	Message sequence number	Message sequence number that matches the value of the SEQUENCE field, as well as any other matches, such as group identifier.

Table 4-4. Using the MATCH options (continued)

MATCH_ option	Retrieves message with specified	Message retrieved must have
MSG_TOKEN	Message token	Message token that matches the value of MSGTOKEN—only for queues that have an index type of MSGTOKEN. You cannot specify other match options with MATCH_MSG_TOKEN.

BROWSE options

For MQ/204 after V6R1.0, there are new options for browsing a queue.

BROWSE_MSG_UNDER_CURSOR, a new option on the MQGET statement, lets you reread the same message. You can combine this option with the new LOCK or UNLOCK options. See Table 4-5 on page 4-19.

- LOCK option lets you lock a message so that another thread cannot browse it. The lock is automatically released at the end of the unit of work, even if there is no explicit UNLOCK. The LOCK option is also valid with BROWSE_FIRST and BROWSE_NEXT.
- UNLOCK option causes the message to be unlocked without retrieving the message.
- Omitting both LOCK and UNLOCK options unlocks the message, but also retrieves it again.

  The following sequence of calls is an example:

```
MQGET BROWSE_FIRST
MQGET BROWSE_NEXT
                  /*until the required message is found
MQGET BROWSE_MSG_UNDER_CURSOR LOCK
              /* to reread the last message and lock it
MQGET BROWSE_MSG_UNDER_CURSOR UNLOCK
/* unlock the same message without retrieving it again
```

Special handling options

ALL_MSGS_AVAILABLE option

You can specify the new ALL_MSGS_ AVAILABLE option on the MQGET statement, and you can set it as a default for the queue by specifying ALL_MSGS_AVAILABLE on the DEFINE QUEUE command or MODIFY QUEUE statement. The default setting is NOT_ALL_MSGS_AVAILABLE.

If you specify ALL_MSGS_AVAILABLE, a message that is part of a group cannot be retrieved from the queue unless all messages in the group are available for retrieval.

MSGTOKEN option

The 16-byte message token, like the message ID, identifies a message as unique. However, unlike the message ID, the message token is not passed between queue managers, so a message token changes as it passes from one queue manager to another queue manager.

You can retrieve the MSGTOKEN %variable option on an MQGET statement. It is also an input field on the MQGET statement, if you also specified the MATCH_MSG_TOKEN option.

NEW_CORREL_ID option

The NEW_CORREL_ID option on the MQPUT and MQPUT1 statements tells the queue manager to generate a new correlation ID for the message. The CORRELID option should not be specified with this field, because the queue manager generates a unique correlation identifier, rather than taking it from the CORRELID field.

SYNCPOINT_IF_PERSISTENT option

The SYNCPOINT_IF_PERSISTENT option is like the SYNCPOINT option, although the request is under transaction control (affected by MQCMIT and MQBACK statements) only if the message is PERSISTENT. This option is valid only on the MQGET statement.

Examples of writing messages and browsing groups

Writing messages to a group in logical order

The following example writes messages to a group in logical order. Note that it is unnecessary to specify SEQUENCE or GROUPID.

```
%QNAME IS STRING LEN 8
%MSG2  IS STRING LEN 100
%MSG1  IS STRING LEN 100
%MSGID IS STRING LEN 24
%COUNT IS FIXED
%LO    = 'LOGICAL_ORDER'
%MSG1  = 'PUTS 4 MESSAGES IN A LOGICALLY ORDERED GROUP'
%COUNT = 0
%GSTAT = 'G'    /? group status for first four messages?/
OPEN QUEUE %QNAME   OUTPUT
REPEAT 3 TIMES
  %COUNT = %COUNT + 1
  %MSG2  = %MSG1 WITH %COUNT
  MQPUT %MSG2 ON %QNAME   GSTATUS %GSTAT LOGICAL_ORDER
  CALL PRINT.STATUS              /? test return codes ?/
END REPEAT
%GSTAT = 'L'     /? group status for last message only ?/
%COUNT = %COUNT + 1
%MSG2 = %MSG1 WITH %COUNT
MQPUT %MSG2 ON %QNAME   GSTATUS %GSTAT  ?%LO
          /? logical order specified as runtime option ?/
```

Writing messages to a group out of sequence

The following example writes messages to a group out of sequence. Notice that we specify NOT_LOGICAL_ORDER, and take care to set SEQUENCE, GSTATUS, and GROUPID correctly.

```
%MSG1  = '5 MESSAGES IN A GROUP OUT OF SEQUENCE'
%COUNT = 0
*             imbed the image definition for the message descriptor
I MQMDV2
PREPARE IMAGE MQMD
%NLO   = 'NOT_LOGICAL_ORDER'
*               for the first physical PUT, the group must be null
%GRP   = ' '
*                   GSTATUS must be L for the last logical message
%GSTAT = 'L'                    /? group status for SEQUENCE = 5 ?/
OPEN QUEUE %QNAME   OUTPUT
%SEQ  = 5
*              we can write the message with sequence number 5 first
```

```
REPEAT 4 TIMES
  %COUNT = %COUNT + 1
  %MSG2  = %MSG1 WITH %COUNT
  *                          specify DESCRIPTOR to retrieve the GROUPID
  MQPUT %MSG2 ON %QNAME  GSTATUS %GSTAT -
   SEQUENCE %SEQ       /? this group in reverse order of sequence ?/ -
   NOT_LOGICAL_ORDER -
   GROUPID %GRP     /? null for first PUT, then generated GROUPID ?/ -
   DESCRIPTOR MQMD          /? needed to retrieve generated GROUPID ?/

   %SEQ   = %SEQ - 1           /? next PUT will have lower SEQUENCE ?/
   CALL PRINT.STATUS                /? always check the return code ?/
   *                          GSTATUS must be G for all other messages
   %GSTAT = 'G'                 /? group status for SEQUENCE 1 to 4 ?/
   %GRP   = $MQMD:GROUPID         /? retrieve the generated GROUPID ?/
  PRINT 'Generated GROUPID IS: ' WITH %GRP
END REPEAT
%COUNT = %COUNT + 1
%MSG2  = %MSG1 WITH %COUNT
MQPUT %MSG2 ON %QNAME   GSTATUS %GSTAT  -
 SEQUENCE %SEQ         /? last PUT has SEQUENCE 1 in this example ?/ -
 ?%NLO GROUPID %GRP          /? use generated GROUPID for last time ?/
CALL PRINT.STATUS
```

Browsing a group of messages

This example browses a group of messages. In this case, we are using the option MATCH_GROUP_ID to select messages in a certain group, and LOGICAL_ORDER to ensure that we retrieve them in sequence number order. Therefore, the index type for this queue must be GROUPID.

```
%QN IS STRING LEN 8
%MG = 'MATCH_GROUP_ID'
%GI = 'GROUPID %GRP'
%LO = 'LOGICAL_ORDER'
INCLUDE MQMD         /? image definition of V2 message descriptor ?/
%COUNT IS FIXED
%TEXT IS STRING LEN 255
%GRP IS STRING LEN 24
%MSGID IS STRING LEN 24
%MSGTOKEN IS STRING LEN 16
%CORRELID IS STRING LEN 24
%GSTAT    IS STRING LEN 1
%SEQ IS FIXED
*                    you can specify LOGICAL_ORDER on OPEN or MODIFY
OPEN QUEUE %QN BROWSE %LO
CALL PRINT.STATUS                 /? always check the return code ?/
%COUNT = 0
MQGET %TEXT FROM %QN   NO_WAIT   -
```

```
 BROWSE_FIRST                -
 DESCRIPTOR MQMD                              /? retrieve group ID ?/
CALL PRINT.STATUS
%GRP      = %MQMD:GROUPID               /? save these values from ?/
%SEQ      = %MQMD:MSGSEQNO              /? the message descriptor ?/
%MSGID    = %MQMD:MSGID
%CORRELID = %MQMD:CORRELID
*                 NOW READ THE WHOLE QUEUE USING THE 'MATCH' KEYWORD
MQGET %TEXT FROM %QN      NO_WAIT   -
 BROWSE_FIRST ?%MG       /? MATCH_GROUP_ID can be a runtime var ?/ -
 ?%GI              /? GROUPID %GRP is another run-time variable ?/ -
 DESCRIPTOR MQMD
CALL PRINT.STATUS
%COUNT = %COUNT + 1
READ.NEXT:
MQGET %TEXT FROM %QN      NO_WAIT  -
 ?%MG ?%GI             -
 BROWSE_NEXT   -
 DESCRIPTOR MQMD GSTATUS %GSTAT
CALL PRINT.STATUS
IF $STATUS = 23 THEN
   JUMP TO END.LOOP       /? end if no more messages in the group ?/
END IF
IF $STATUS = 12 THEN
   JUMP TO END.LOOP                      /? test for other errors ?/
END IF
%COUNT = %COUNT + 1                /? count messages in the group ?/
*     It would also be possible to test for %GSTAT = 'L' to test for
*                                       the end of the loop, as here:
IF %GSTAT = 'L'                         /? last message in the group ?/
 JUMP TO END.LOOP         /? end if no more messages in the group ?/
END IF
JUMP TO READ.NEXT
END.LOOP:
*   $STATUSD = 2247 if the Queue has wrong index type
IF $STATUSD = 2247 THEN
   AUDIT 'WRONG TYPE OF INDEX FOR MATCH'
ELSE
   AUDIT ' Messages in the group: ' WITH %COUNT
END IF
CLOSE QUEUE %QN
```

Supporting Java messages with the RFH2 keyword

The RFH2=(*image* | BUFFER) option of the MQGET and MQPUT statements support Java messages and the MQRFH2 header. The following example includes User Language statements using the RFH2 keyword.

```
WRITE IMAGE MQRFH2 ON BUFFER POSITION 1
```

```
*             the RFH2 header is now in the buffer
MQPUT DATA1  ON %QNAME   GSTATUS %GSTAT -
 LOGICAL_ORDER RFH2 BUFFER -
 GROUPID %GRP  DESCRIPTOR MQMD
*  The following MQGET reads the RFH2 header into the
*  image called MQRFH2, and the data into the %variable
*  called %TEXT
MQGET %TEXT FROM %QNAME NO_WAIT    -
 DESCRIPTOR MQMD -
 RFH2 MQRFH2
```

See “MQGET statement” on page 4-26 and “MQPUT statement” on page 4-34 for syntax layout and more details.

Updating the Version 2 message descriptor (MQMD V2)

The image definition for the message descriptor, which you can retrieve using the DESCRIPTOR option, is updated to include the new fields in MQMD Version 2. The following sample image can be specified as target of the WebSphere MQ MQGET DESCRIPTOR option. It matches the layout of the WebSphere MQ MQMD data structure that is described in the *WebSphere MQ Application Programming Reference.*

```
 IMAGE MQMD
* THIS IS THE VERSION 2 MQMD
STRUCID     IS STRING LEN 4
* TYPE OF STRUCTURE
VERSION     IS BINARY LEN 4
* VERSION NUMBER OF THE STRUCTURE
REPORT      IS BINARY LEN 4
* OPTIONS FOR REPORT MESSAGES
MSGTYPE     IS BINARY LEN 4
* THE TYPE OF MESSAGE
* (1=REQUEST, 2=REPLY, 4=REPORT, 8=DATAGRAM)
EXPIRY      IS BINARY LEN 4
* MESSAGE LIFETIME
FEEDBACK    IS BINARY LEN 4
* FEEDBACK CODE
ENCODING    IS BINARY LEN 4
* DATA ENCODING
CODECHARSETID  IS BINARY LEN 4
* CODED CHARACTER SET IDENTIFIER
FORMAT         IS STRING LEN 8
* FORMAT NAME
PRIORITY       IS BINARY LEN 4
* MESSAGE PRIORITY
PERSISTENCE    IS BINARY LEN 4
* MESSAGE PERSISTENCE
MSGID          IS STRING LEN 24
* MESSAGE IDENTIFIER
```

```
CORRELID       IS STRING LEN 24
* CORRELATION IDENTIFIER
BACKOUTCOUNT   IS BINARY LEN 4
* BACKOUT COUNTER
REPLYTOQ       IS STRING LEN 48
* NAME OF REPLY QUEUE FOR REQUESTS
REPLYTOMGR     IS STRING LEN 48
* NAME OF REPLY QUEUE MANAGER FOR REQUESTS
USERIDENTIFIER IS STRING LEN 12
* USER IDENTIFIER
ACCOUNTINGTOKEN  IS STRING LEN 32
* ACCOUNTING TOKEN
APPLIDENTITYDATA IS STRING LEN 32
* APPL DATA RELATING TO IDENTITY
PUTAPPLTYPE      IS BINARY LEN 4
* TYPE OF APPLICATION THAT PUT THE MSG
PUTAPPLNAME      IS STRING LEN 28
* ID OF APPLICATION THAT PUT THE MESSAGE
PUTDATE          IS STRING LEN 8
* DATE WHEN MESSAGE WAS PUT
PUTTIME          IS STRING LEN 8
* TIME WHEN MESSAGE WAS PUT
APPLORIGINDATA   IS STRING LEN 4
* APPLICATION DATA RELATING TO ORIGIN
GROUPID          IS STRING LEN 24
* GROUP ID
MSGSEQNO         IS BINARY LEN 4
* MESSAGE SEQUENCE NUMBER WITHIN GROUP
OFFSET           IS BINARY LEN 4
* OFFSET OF SEGMENT WITHIN MESSAGE
MSGFLAGS         IS BINARY LEN 4
* MESSAGE FLAGS
ORGLEN           IS BINARY LEN 4
* ORIGINAL LENGTH
END IMAGE
```

MQRFH2 image format

You can use the following image when there are no variable length fields.

```
IMAGE MQRFH2
* THIS IS THE RFH2 HEADER FOR JAVA
STRUCID     IS STRING LEN 4
* TYPE OF STRUCTURE'RFH '
VERSION     IS BINARY LEN 4
* VERSION NUMBER OF THE STRUCTURE - 2
LENGTH      IS BINARY LEN 4
* LENGTH OF THE STRUCTURE (36)
ENCODING    IS BINARY LEN 4
```

```
* THE TYPE OF ENCODING
CHARSET     IS BINARY LEN 4
* CODEDCHARSETID OF DATA
FORMAT      IS STRING LEN 8
* FORMAT OF DATA FOLLOWING RFH2
FLAGS       IS BINARY LEN 4
* FLAGS - SET TO 0
NAMEVALUECCSID IS BINARY LEN 4
* 1208, 1200, 13488, OR 17584
* CODEDCHARSETID OF VARIABLE PART
END IMAGE
```

Command and statements

The options in Table 4-5 handle grouping messages.

Table 4-5. Options for grouping messages in MQ/204

Option	Purpose	Discussion	Can be set on
ALL_MSGS_ AVAILABLE	Use on an MQGET call to retrieve only messages that belong to complete groups, as well as messages that do not belong to a group.		DEFINE QUEUE MODIFY QUEUE MQGET OPEN QUEUE
NOT_ALL_MSGS_ AVAILABLE (default)	Use on an MQGET call to retrieve any message in the queue, including partial groups.		DEFINE QUEUE MODIFY QUEUE MQGET OPEN QUEUE
BROWSE_MSG_ UNDER_CURSOR	Reread the same message again.	You would typically do this to unlock a locked message or vice versa. See the LOCK and UNLOCK options later in this table.	MQGET
GROUPID	The GROUPID is a 24-byte field that you specify when writing messages that belong to a group. You must specify GROUPID as a null string (or spaces) on the first message of any group. See “Message groups” on page 4-3 for a more detailed discussion of GROUPID.	For MQGET, MQPUT, and MQPUT1 calls, the group identifier of the message is returned in the MQMD, not in the GROUPID field, which is always an input field only. On an MQGET call, this is an input field when MATCH_GROUP_ID is specified.	MQGET MQPUT MQPUT1
GSTATUS = [' ' | 'G' | 'L']	The one-byte GSTATUS field determines whether the message is part of a group. The possible values are: • ’G’ (all messages in a group except the last or only one) • ’L’ (the last or only logical message in a group) • Space or null (the message is not in a group)	On MQGET operations, this is an output %variable. If you are writing messages that belong to a group but are not in logical order, set this to ’L’ for the message that has the highest sequence number. The default, for MQPUT and MQPUT1 operations, is null—the message does not form part of a group.	MQGET MQPUT MQPUT1

Table 4-5. Options for grouping messages in MQ/204 (continued)

Option	Purpose	Discussion	Can be set on
LOCK	LOCK lets you browse a queue and lock the current message so that other threads do not see it.	The default is not to lock messages while browsing them.	MQGET
UNLOCK	UNLOCK lets you remove a lock from a previously-locked message.	When you specify UNLOCK, the message itself is not returned, it is just unlocked.	MQGET
LOGICAL_ORDER	Write one group at a time and write messages in sequential order, starting at 1.	This is the simpler way of writing a group of messages. However, it is not the default.	DEFINE QUEUE MODIFY QUEUE MQGET MQPUT OPEN QUEUE
NOT_LOGICAL_ ORDER (default)	This option gives you more control over the writing of groups, but requires that you specify SEQUENCE and GROUPID for each message.	You can also use this setting for messages that are not a group.	DEFINE QUEUE MODIFY QUEUE MQGET MQPUT MQPUT1 OPEN QUEUE
MATCH_CORREL _ID (default)	Retrieve messages that match the specified CORRELID.	If CORRELID is not specified, this option is ignored.	MQGET
NOT_MATCH_ CORREL_ID	Retrieve any message in the queue, if CORRELID is specified.		MQGET
MATCH_MSG_ GROUP_ID	Retrieve messages that match the specified GROUPID.	The queue must be indexed by GROUPID.	MQGET
MATCH_MSG_ID (default)	Retrieve messages that match the specified MSGID.	If MSGID is not specified, this option is ignored.	MQGET
NOT_MATCH_ MSG_ID	Retrieve any message in the queue, if MSGID is specified.		MQGET
MATCH_MSG_ SEQ_NUMBER	Retrieve the message that matches SEQUENCE for the current group. If there is no current group, the only valid value of SEQUENCE is 1.	The queue must be indexed by GROUPID.	MQGET
MATCH_MSG_ TOKEN	Use this to retrieve messages that match the specified MSGTOKEN	The queue must be indexed by MSGTOKEN.	MQGET

Table 4-5. Options for grouping messages in MQ/204 (continued)

Option	Purpose	Discussion	Can be set on
MSGTOKEN	You can select messages by 16-byte message token (MSGTOKEN) by specifying this field as an input field on MQGET statements with MATCH_MSG_TOKEN.	This is also an output field, retrieving the message token that was assigned by the queue manager. (On the MQGET call, it can also be an input field, if MATCH_MSG_TOKEN is specified.	MQGET MQPUT MQPUT1
NEW_CORREL_ID	This option on the MQPUT statement tells the queue manager to generate a new correlation ID for the message.	The CORRELID option should not be specified with this option, because the queue manager generates a unique correlation identifier, rather than taking it from the CORRELID option.	MQPUT MQPUT1
SEQUENCE =%variable	For messages that are part of a group and NOT_LOGICAL_ORDER is specified (or is the default), this number specifies the logical position of the message within the group, which need not be the physical sequence in which the messages are written. The logical first message is message sequence 1. On MQGET operations, this is an input variable used only with the MATCH_MSG_SEQ_ NUMBER option. On MQPUT and MQPUT1 operations, this is an input variable that specifies the logical position within the group of the message being written.	This is a fixed %variable. To retrieve the logical sequence of a message in an MQGET operation, use the DESCRIPTOR option and get the sequence number from the returned MQMD image. When using SEQUENCE with the MATCH_MSG_SEQ_ NUMBER option, you can only match a SEQUENCE greater than 1 within the current group. You cannot, for example, do a repeated BROWSE_NEXT with SEQUENCE 2 to jump from one group to another.	MQGET MQPUT MQPUT1
SYNCPOINT_IF_ PERSISTENT	This option on the MQGET statement works like SYNCPOINT, but is effective only for persistent messages.		MQGET

DEFINE QUEUE: Identifying a WebSphere MQ queue

Privileges System manager or User 0

Function Identifies an z/OS-WebSphere MQ queue in a queue manager where applications can put (MQPUT and MQPUT1) and get (MQGET) messages.

Syntax

```
DEFINE {QUEUE | Q} name [LIKE previousname] WITH

SCOPE=SYSTEM {QUEUEMANAGER | QM}=qmanentityname

[{QUEUENAME | QNAME}=externalqueuename] [options…]
```

Where

- *name* identifies the queue referred to in the Online. name must be:
  - Unique, or an error is issued.
  - 1-48 characters long and begin with a letter, followed by letters, numbers, periods, or underscores.
- QUEUEMANAGER (or QM) clause names the queue manager for the queue, and must refer to a queue manager previously defined by the WebSphere MQ administrator.

  Use the optional QUEUENAME (or QNAME) clause, if the external name of the queue (as defined to WebSphere MQ) differs from the name you refer to in the Online. External queue names can be up to 48 characters in length. The external queue name is validated when the queue is actually opened. If you omit QUEUENAME (or QNAME), the external name of the queue is assumed to be name.
- *options* are default characteristics that are used to manipulate the queues.
  - ACCEPT_TRUNCATED_MSG or NO_ACCEPT_TRUNCATED_MSG (default)
  - ALL_MSGS_AVAILABLE or NOT_ALL_MSGS_AVAILABLE (default)
  - FAIL_IF_QUIESCING or NO_FAIL_IF_QUIESCING (default)
  - CONTEXT
  - EXPIRY
  - FORMAT
  - LOGICAL_ORDER or NOT_LOGICAL_ORDER (default)
  - MSGTYPE
  - PERSISTENT, NOT_PERSISTENT, or PERSISTENCE_AS_Q_DEF (default)
  - PRIORITY or PRIORITY_AS_Q_DEF (default)
  - REPLY_QMGR
  - REPLY_QUEUE

- REPORT
- SYNCPOINT (default) or NO_SYNCPOINT
- WAIT or NO_WAIT (default)
- WAIT_TIME

Usage The queue definition has system scope.

Note: Remote queues do not require a DEFINE QUEUE command. Local dynamic queues require a DEFINE QUEUE command only if they were not created by MQ/204 in the current Online job.

Defining local dynamic queues

When a local dynamic queue is created using an MQOPEN statement, it is created with SCOPE=SYSTEM. Model 204 sets up control blocks as if you had defined the queue using the DEFINE command.

If the Online is cycled and the permanent local dynamic queue still exists, to access it in a subsequent run, you must issue a DEFINE QUEUE command with the full name of the local dynamic queue, even if the queue manager created the name when the local dynamic queue was originally defined.

Security considerations

Applications creating local dynamic queues via the OPEN QUEUE statement cannot reference model queues unless the system manager defines the model queues.

MODIFY QUEUE statement

Function Alters the current default option(s) for an open queue for the issuing user.

Syntax

```
MODIFY QUEUE {%variable | name |
  external_qmanager:external_queue}
  {[option...] [?%variable...]}
```

Where

- Queue to operate on is specified as:

Queue as…	Specifies
%variable	Queue name
name	Literal without quotation marks
external_qmanager:external_queue	External name of a queue manager that contains the remote queue to process and the external name of the remote queue

- *option* is one or more of the following options:
  - ACCEPT_TRUNCATED_MSG or NO_ACCEPT_TRUNCATED_MSG
  - ALL_MSGS_AVAILABLE or NOT_ALL_MSGS_AVAILABLE (default)
  - CONTEXT
  - EXPIRY
  - FAIL_IF_QUIESCING or NO_FAIL_IF_QUIESCING
  - FORMAT
  - LOGICAL_ORDER or NOT_LOGICAL_ORDER (default)
  - MSGTYPE
  - PERSISTENT or NOT_PERSISTENT
  - PERSISTENCE_AS_Q_DEF
  - SYNCPOINT or NO_SYNCPOINT
  - PRIORITY or PRIORITY_AS_Q_DEF
  - REPLY_QMGR
  - REPLY_QUEUE
  - REPORT
  - WAIT or NO_WAIT
  - WAIT_TIME

- *?%variable* specifies option(s) of the MODIFY QUEUE statement to compile at evaluation time. For more information on ?%variables, see *Model 204 MQ/204 Reference Manual*.

Usage

Options that apply to MQGET, MQPUT, and MQPUT1 statements can be specified in the DEFINE QUEUE command, and defaults for MQGET and MQPUT statements can be specified in an OPEN QUEUE statement.

You can override defined options by specifying the desired options directly on an MQGET, MQPUT, or MQPUT1 statement.

If you are repeatedly overriding DEFINE QUEUE command options, you might prefer to alter the current options for an open queue using the MODIFY QUEUE statement.

MQGET statement

Function Retrieves a message from a currently open queue.

Syntax

```
MQGET {(image[,image]...) | %variable | BUFFER
      | MQ_BUFFER}
      [RFH2=(image | BUFFER)]
      [FROM] {%qvariable | entname
      | external_queuemanager:external_queue}
      [BUFLEN={%bvar | n}]
      [MSGLEN=%mvar]
      [[option...] [?%variable...]]
```

Where

- *image* or *%variable* or *BUFFER* (formerly and still accepted MQ_BUFFER) specifies the target into which the message data is placed. You can specify up to 10 images.

If the target is…	Then…
Image	Image item READLEN is filled in with the number of bytes of message data read into the image
%variable	Number of bytes read can be obtained with $LEN
BUFFER (formerly and still accepted MQ_BUFFER)	Message is simply left in the Universal Buffer, where is remains intact until the next MQ/204 statement is issued
RFH2=(image | BUFFER)	Java RFH2 header can be accommodated.

- FROM clause identifies the queue to operate on as specified:

Queue as…	Specifies
%qvariable	Queue name
entname	Literal without quotation marks
external_qmanager:external_queue	External name of a queue manager that contains the remote queue to process and the external name of the remote queue

- BUFLEN can determine the destination size of BUFFER area in bytes, as follows:
  - If BUFLEN was specified and the size of your BUFFER area is less

than BUFLEN, BUFFER area is resized to the value of BUFLEN. Otherwise, BUFFER size is unchanged.

– If BUFLEN was not specified:

Destination byte size for…	Is…
String %variable	Declared length of the %variable.
Image	Maximum length of the image.
Multiple images	Sum of the individual image sizes.

If the size of your BUFFER area is less than the destination size, the existing BUFFER area is deleted and a new BUFFER area allocated with a size equal to the destination size.

- *MSGLEN=%mvar* specifies the actual length of the message.
- *option* is one or more of the following options:
  – ACCEPT_TRUNCATED_MSG or NO_ACCEPT_TRUNCATED_MSG
  – ALL_MSGS_AVAILABLE_ or NOT_ALL_MSGS_AVAILABLE (default)
  – BROWSE_FIRST
  – BROWSE_MSG_UNDER_CURSOR
  – BROWSE_NEXT
  – CONVERT or NO_CONVERT
  – CORRELID=value
  – DESCRIPTOR=image
  – FAIL_IF_QUIESCING or NO_FAIL_IF_QUIESCING
  – GROUPID=%variable
  – GSTATUS=%variable
  – LOCK or UNLOCK
  – LOGICAL_ORDER or NOT_LOGICAL_ORDER (default)
  – MATCH_CORREL_ID (default) or NOT_MATCH_CORREL_ID
  – MATCH_GROUP_ID
  – MATCH_MSG_ID (default) or NOT_MATCH_MSG_ID
  – MATCH_MSG_SEQ_NUMBER
  – MATCH_MSG_TOKEN
  – MARK_SKIP_BACKOUT
  – MSGID=value
  – MSGTOKEN=%variable
  – MSG_UNDER_CURSOR
  – NEW_CORREL_ID

  - SEQUENCE=%variable
  - SYNCPOINT or NO_SYNCPOINT or SYNCPOINT_IF_PERSISTENT
  - WAIT or NO_WAIT
  - WAIT_TIME=value
- ?%variable specifies option(s) of the MQGET statement to compile at evaluation time.

Analyzing an MQGET statement

When you issue an MQ/204 MQGET statement, the WebSphere MQ MQGET is issued. WebSphere MQ places a message, or part of a message, in your BUFFER area and returns:

  - Actual length of the message that was retrieved or partially retrieved
  - Completion code and reason code that qualifies the completion code
- The length of the message is recorded as follows:

Argument or function	Is set to
$BUFFER_USED	Length of the data retrieved into BUFFER area
$MQ_MESSAGE_LEN	Actual length of the message
MSGLEN=%mvar	Actual length of the message

- If the completion code indicates successful completion:

If the destination is…	Then…
BUFFER area	$BUFFER_POSITION is set to 1.
Anything other than BUFFER area	Message is copied from BUFFER area to the destination.
Image or images	Each READLEN is set.

$BUFFER_POSITION is set to the byte after the last byte copied.

- If the completion code indicates unsuccessful completion:
  - $STATUS is set appropriately
  - $STATUSD is set to the WebSphere MQ reason code
  - $BUFFER_POSITION is set to 1
  - No further processing is done

Using the BUFFER area

Data transfers between WebSphere MQ and Model 204 are performed via the BUFFER area. You can perform all MQ/204 operations without directly

referencing the BUFFER area. However, for maximum flexibility, you can directly manipulate the contents of the BUFFER area using the following statement forms:

- MQGET BUFFER
- MQPUT BUFFER
- READ IMAGE FROM BUFFER
- WRITE IMAGE ON BUFFER

Each user thread of MQ/204 has only one BUFFER area, which is private to that thread. BUFFER area resides in a separately allocated area of memory; it does not occupy space in the user's server, nor in CCATEMP. BUFFER area is allocated automatically for you when required, and decommissioned when you log off.

When an MQGET statement retrieves a message from WebSphere MQ to a %variable, image, or image list, the data is first moved from WebSphere MQ to BUFFER area, and then copied to the %variable, image, or image list. When an MQPUT or MQPUT1 statement sends a message from a %variable, image, or image list to WebSphere MQ, the data is first copied from the %variable, image, or image list to BUFFER area and then moved to WebSphere MQ.

Handling the end of a request

- Under the usual circumstances, at request end, the user's BUFFER area remains allocated but its contents are cleared.
- In an APSY subsystem with AUTOCOMMIT=NO, the user’s BUFFER area remains allocated and its contents are unchanged.

Usage

Options in common with OPEN QUEUE and DEFINE QUEUE statements inherit default values from the OPEN QUEUE statement, which in turn inherits from the queue:

- If DESCRIPTOR=image is specified, the MQMD message descriptor is returned, even if the MQGET failed. However, if the MQGET failed, the contents of image may or may not be meaningful, depending on the nature of the failure.
- After a get operation with BUFFER, the data can be loaded into an image with an extension to the READ IMAGE statement.
- When a local dynamic queue is deleted, any MQGET statements with the WAIT option that are outstanding against the queue are canceled and reason code MQRC_Q_DELETED is returned.

Using the browse options

On queues open for browsing only, if neither BROWSE_FIRST nor BROWSE_NEXT is specified, then the default operation of MQGET is BROWSE_FIRST on the first MQGET after the OPEN QUEUE and BROWSE_NEXT on all subsequent MQGET statements.

On queues open for both browsing and input, if neither BROWSE_FIRST nor BROWSE_NEXT is specified, then, as the MQGET statement retrieves messages, they are deleted from the queue. This is consistent with the behavior of the WebSphere MQ API.

- On get operations, the READLEN item of each image is set. If the data runs out before filling all the images, then READLEN is set to zero for the remaining images.
- If BUFLEN is specified, BUFLEN is used to limit the number of bytes retrieved from WebSphere MQ.
- MSGLEN is an output %variable; its value before the MQGET is irrelevant.

BROWSE_MSG_UNDER_CURSOR will reread the current message. If it was locked before, it will be unlocked unless LOCK is specified again.

Truncated messages

Truncation of messages can occur at two different points:

- When the data is moved from WebSphere MQ to the BUFFER area ($STATUS=12):

  ```
  ($MQ_MESSAGE_LEN > $BUFFER_USED)
  ```

- When the data is copied from the BUFFER area to the destination ($STATUS=14):

  ```
  ($MQ_MESSAGE_LEN = $BUFFER_USED)
  ```

If truncation occurs when the data is moved from WebSphere MQ to the BUFFER area, the setting of the ACCEPT_TRUNCATED_MSG option determines whether or not WebSphere MQ leaves the message on the queue; or, if in browse mode, whether or not WebSphere MQ advances the browse cursor.

Error handling consideration

MQGET statement processing always retrieves the message from WebSphere MQ into the user's BUFFER area.

- If the WebSphere MQ call fails, $STATUS is set to 12, $STATUSD is set to the WebSphere MQ reason code, and the operation ends.

- If the call succeeds, the message is copied from the BUFFER area into the target—%variable, image, or image list.
- If the target is not large enough to hold the message retrieved, $STATUS is set to 14 to indicate truncation; otherwise, $STATUS is set to 0 to indicate success.

Under some circumstances, the WebSphere MQ call can fail but still return a message or part of a message. In those cases, $STATUS is set to 12, $STATUSD is set to the WebSphere MQ reason code, but the operation continues and the message is copied from the BUFFER area into the target.

If the target is not big enough to hold the message (or partial message) retrieved, $STATUS is not reset to 14, but instead retains its original value. In other words, $STATUS and $STATUSD always report the first error, and the subsequent error is not explicitly reported.

The circumstances under which a WebSphere MQ MQGET statement can fail, but still return a message or part of a message are:

WebSphere MQ reason code	**Message**
2079	TRUNCATED_MSG_ACCEPTED
2080	TRUNCATED_MSG_FAILED
2110-2120	Various conversion errors

If an MQGET statement returns $STATUS=12 and $STATUSD=2079, or 2080, or 2110–2120, and the target is a %variable, image or image list, then the program *must* determine whether the message was truncated when copied to the target. You cannot use $STATUS=14.

You can use the following after an initial error to determine if truncation also occurred:

```
  IF ($BUFFER_POSITION LE $BUFFER_USED) THEN
* We truncated moving from the buffer to the %variable
* or image target
  ELSE
* We have $BUFFER_POSITION = $BUFFER_USED + 1
* We did not truncate moving the buffer to the %variable
* or image target
  END IF
```

Note: In the case of $STATUSD=2079 or 2080, the message was truncated when moved from WebSphere MQ to the BUFFER area, but could not have been truncated again when copied from the BUFFER area to the target.

Handling an incoming message with an RFH2 header

If you expect the incoming message to contain an RFH2 header, use the RFH2=(*image* | BUFFER) option. You can specify an image into which the header will be copied, or you can specify BUFFER, in which case the RFH2 header is left in the BUFFER, preceding the data.

The RFH2 header, if the target is an image, is copied from the buffer to an image, but the RFH2 header is not removed from the buffer—it still precedes the data.

- If the target for the RFH2 header specified BUFFER, the RFH2 header is simply left at the beginning of the buffer.
- If the target of the data is specified as an image or %variable, the data is copied from its position behind the RFH2 header (if present) to the image or %variable.
- If the target of the data is specified BUFFER, it is left in place in the buffer behind the RFH2 header.

In all cases where the RFH2 header is involved, the buffer after the operation contains the RFH2 header followed by the data, and $BUFFER_POSITION variable has the position (=offset+1) of the data, if the data is left in the buffer, or beyond the data if the data has been copied to an image, image list or variable.

The $BUFFER_USED has the total length of RFH2 header and data.

Using RFH2 keyword with MQGET

In all cases of GET, MSGLEN is set to the length of the data received, excluding the length of the RFH2 header if any.

GET target	RFH2 in image	RFH2 in buffer
Data in image or %variable	$BUFFER_USED set to length of RFH2 header, plus length of data. $MQ_MESSAGE_LEN set to length of data. $BUFFER_POSITION set to byte beyond last byte of data copied. RFH2 copied to image. Data copied to image or %variable.	$BUFFER_USED set to length of RFH2 header, plus length of data. $MQ_MESSAGE_LEN set to length of data.$BUFFER_POSITION set to byte beyond last byte of data copied. Data copied to image or %variable.

GET target	RFH2 in image	RFH2 in buffer
Data in buffer	$BUFFER_USED set to length of RFH2 header + length of data. $MQ_MESSAGE_LEN set to length of data. $BUFFER_POSITION set to first byte of data (the byte beyond the RFH2 header). RFH2 copied to image.	$BUFFER_USED set to length of RFH2 header + length of data. $MQ_MESSAGE_LEN set to length of data. $BUFFER_POSITION set to first byte of data (the byte beyond the RFH2 header).

MQPUT statement

Function Places a message on a currently open queue.

Syntax

```
MQPUT {(image[,image1]...) | %variable | 'string'
  | BUFFER | MQ_BUFFER}
  [RFH2=(image | BUFFER)]
  [ON] {%qvariable | entname
  | external_qmanager:external_queue}
  [BUFLEN= {%bvar | n}]
  [[option...] [?%variable...]]
```

Where

- *image, %variable, 'string'* (quoted literal), or BUFFER (formerly, and still accepted, MQ_BUFFER) area specified contains the message data.

This message source…	Specifies…
image	To swap images in and process serially. You can specify up to 10 images.
%variable	Current value of variable.
'string'	Literal with quotation marks.
BUFFER (formerly, and still accepted as MQ_BUFFER)	Message is preloaded in the Universal Buffer area.
RFH2=(image | BUFFER)	Java RFH2 header can be accommodated.

- Queue to operate on is:

Queue as…	Specifies...
%qvariable	Queue name.
entname	Literal without quotation marks.
external_qmanager:external_queue	External name of a queue manager that contains the remote queue to process and the external name of the remote queue.

- BUFLEN specifies in bytes, %bvar or n, the size of BUFFER area.
- *option* is one or more of the following options:
  - ACCOUNTINGTOKEN
  - APPLIDENTITYDATA=value

- APPLORIGINDATA=value
- CONTEXT=value or NO_CONTEXT
- CORRELID=value
- DESCRIPTOR=image
- DEFAULT_CONTEXT
- EXPIRY=value
- FAIL_IF_QUIESCING or NO_FAIL_IF_QUIESCING
- FEEDBACK=value
- FORMAT=value
- GROUPID=%variable
- GSTATUS=%variable
- LOGICAL_ORDER or NOT_LOGICAL_ORDER (default)
- MSGID=value
- MSGTYPE=value
- PASS_ALL_CONTEXT
- PASS_IDENTITY_CONTEXT
- PERSISTENT or NOT_PERSISTENT
- PERSISTENCE_AS_Q_DEF
- PRIORITY=value or PRIORITY_AS_Q_DEF
- PUTAPPLNAME=value
- PUTAPPLTYPE=value
- PUTDATE=value
- PUTTIME=value
- REPLY_QMGR=name
- REPLY_QUEUE=name
- REPORT=options
- SEQUENCE=%variable
- SET_ALL_CONTEXT
- SET_IDENTITY_CONTEXT
- SYNCPOINT or NO_SYNCPOINT
- USERIDENTIFIER='value'

?%variable specifies option(s) of the MQPUT and MQPUT1 statements to compile at evaluation time.

Usage

If an MQPUT or MQPUT1 statement sends a message from a %variable, image, or image list to WebSphere MQ, the data is first copied from the

%variable, image, or image list to the BUFFER area, and then copied to WebSphere MQ.

MQPUT and MQPUT1 processing

When WebSphere MQ MQPUT or MQPUT1 is issued, WebSphere MQ:

- Copies the message from the user's BUFFER area
- Returns a completion code and reason code (qualifies the completion code)

Irrespective of the completion code:

- $MQ_MESSAGE_LEN is set to the length of the message
- $BUFFER_POSITION is set to 1

If the completion code does not indicate successful completion, $STATUS is set appropriately, and $STATUSD is set to the WebSphere MQ reason code.

Usage notes for options

- Options in common with the DEFINE QUEUE command and the OPEN QUEUE statement inherit default values from the OPEN QUEUE statement, which in turn inherits from the queue.
- You can specify the following identity context options only if you also specify the SET_ALL_CONTEXT or SET_IDENTITY_CONTEXT option:
  - ACCOUNTINGTOKEN
  - APPLIDENTITYDATA
  - USERIDENTIFIER
- You can specify the following origin, context parameters only if you also specify the SET_ALL_CONTEXT option:
  - APPLORIGINDATA
  - PUTAPPLNAME
  - PUTAPPLTYPE
  - PUTDATE
  - PUTTIME
- If DESCRIPTOR=image is specified, the MQMD message descriptor is returned, even if the MQPUT or MQPUT1 failed. However, if the MQPUT or MQPUT1 failed, the contents of image might or might not be meaningful, depending on the nature of the failure.
- On each MQPUT statement, the message context is set by specifying one of the following, mutually exclusive, options:
  - DEFAULT_CONTEXT

- NO_CONTEXT
- PASS_ALL_CONTEXT
- PASS_IDENTITY_CONTEXT
- SET_ALL_CONTEXT
- SET_IDENTITY_CONTEXT

If none of the previous options is specified, WebSphere MQ applies its own default, which is DEFAULT_CONTEXT. For more information on message context, see “Applying date/time stamps to messages” on page 4-38.

Managing BUFFER area

If the message source is BUFFER area:

If…	Then…
$BUFFER_USED=0	Operation returns $STATUS=26
BUFLEN=%bvar was not specified	Message length is $BUFFER_USED
BUFLEN=%bvar was specified	Message length is the minimum of $BUFFER_USED and %lvar

If the source is anything other than BUFFER area, the source size is determined:

For	Size is
String %variable	Current length of the %variable
Image	Maximum length of the image
Multiple images	Sum of the individual image sizes

If the size of the user’s BUFFER area is less than the message length, the existing BUFFER area is deleted and a new BUFFER area allocated with a size equal to the message length.

The message is copied from the source to the BUFFER area.

If BUFLEN=%bvar was…	Message length is…
Not specified	Source size
Specified	Minimum of the source size and %lvar

$BUFFER_USED is set to the message length.

Applying date/time stamps to messages

Each WebSphere MQ message is associated with two origin, context fields called PUTDATE and PUTTIME, which act as a date-stamp and time-stamp, respectively, when a message is posted to a queue.

When a message is put on a queue, these fields are usually filled in. Any WebSphere MQ-enabled program issuing a WebSphere MQ MQPUT statement can explicitly set values for these fields, or alternatively (and most commonly), the values can be set by the WebSphere MQ queue manager. In MQ/204, you explicitly set values for these fields by specifying the option PUTDATE and/or PUTTIME on the MQPUT and MQPUT1 User Language statements. The queue manager sets values for those fields, if you do not specify those options.

When a message is received from a queue, the values of these fields that were set by the message originator are passed in the WebSphere MQ message descriptor, MQMD. In MQ/204, you can obtain a copy of the message descriptor by specifying option DESCRIPTOR=*imagname* on the MQGET User Language statement. The MQMD is not available to you, if you do not specify this option.

If you decide to set or inspect the values of the PUTDATE and PUTTIME fields in your own application, read the IBM manual *WebSphere MQ Application Programming Reference* to understand the format of these fields when their values are set by a WebSphere MQ queue manager. In particular, according to this manual:

> "Greenwich Mean Time (GMT) is used for the PutDate and PutTime fields, subject to the system clock being set accurately to GMT. On OS/2, the queue manager uses the TZ environment variable to calculate GMT."

If you set PUTDATE and PUTTIME in your own applications, WebSphere MQ lets you use any date or time format, any time zone, or even invalid values. CCA strongly recommends, however, that you adhere to the default format used by WebSphere MQ.

Handling an outgoing message with an RFH2 header

Use the RFH2 option if you want the outgoing message to include an RFH2 header.

You can specify an image from which the header will be copied, or you can specify BUFFER, in which case the RFH2 header must be at the start of the BUFFER (position = 1), ahead of the data if that is also sourced from the BUFFER.

- If the RFH2 keyword is present, the FORMAT field will be set to MQHRF2.
- If the RFH2 keyword is present, but there is no valid RFH header in the specified image or buffer, the statement will fail with $STATUS = 47.

The RFH2 header, if specified, is taken from an image and placed in the buffer, or, if BUFFER is specified, is presumed to be at position 1 in the buffer (this will be checked).

- The data, if sourced from an image or %variable, is copied to the buffer so that it immediately follows the RFH2.
- If the data is sourced from the buffer, it is presumed to follow the RFH2 (if the RFH2 header is also sourced from the buffer), or is copied so that it follows the RFH2 header (if the RFH2 header is sourced from an image).

In all cases where the RFH2 header is involved, the buffer after the operation contains the RFH2 header followed by the data.

Using RFH2 keyword with MQPUT

PUT source	RFH2 from image	RFH2 from buffer
Data from image or %variable	The RFH2 is copied to the buffer. Then the data is appended to the RFH2 header in the buffer.	The beginning of the buffer is checked for a valid RFH2 header. The data is appended to the RFH2 header in the buffer.
Data from buffer	The data is copied downwards in the buffer to make way for the RFH2 header, which is then copied to the beginning of the buffer.	The RFH2 is assumed to precede the data in the buffer. The beginning of the buffer is checked for a valid RFH2 header.

In all cases of PUT:

- $BUFFER_POSITION is ignored, and set to 1 after the operation.
- $BUFFER_USED is set to the length of RFH2 header, plus the length of data.
- $MQ_MESSAGE_LEN set to length of data.
- MSGLEN set to length of data.
- $BUFFER_POSITION set to first byte of data beyond the RFH2 header.

MQPUT1 statement

Function

Places one message on a queue that is not currently open. The MQPUT1 statement is a combination of OPEN QUEUE, MQPUT, and CLOSE QUEUE. It is the most efficient way to put a single message on a queue.

Syntax

```
MQPUT1 {(image [,image1]) | %variable | 'string'
  | BUFFER | MQ_BUFFER}
 [ON] {%qvariable | entname
 | external_qmanager:external_queue} [option…]
```

Where

- *image, %variable, 'string'*, or BUFFER (formerly, and still accepted, MQ_BUFFER) area specified contains the message data.

This message source…	Specifies…
image	To swap images in and process serially. You can specify up to 10 images.
%variable	Current value of variable.
'string'	Literal with quotation marks.
BUFFER (or MQ_BUFFER) area	Message is preloaded in the Universal Buffer area.

- Queue to operate on is specified as:

Queue as…	Specifies...
%qvariable	Queue name.
entname	Literal without quotation marks.
external_qmanager:external_queue	External name of a queue manager that contains the remote queue to process and the external name of the remote queue.

- *option* is one or more of the options.

  Options are the same as those for MQPUT with one addition: the PASS_USER_CONTEXT option.

  Also, LOGICAL_ORDER is not a permitted option for MQPUT1.

  MQPUT1 inherits options directly from the queue, because it does its own open and is completely independent from the OPEN QUEUE statement.

Usage

The security considerations apply to MQPUT1 as well.

- If the queue named in an MQPUT1 statement is already open, the statement is processed regardless; there is no error.
- On each MQPUT1 statement, the message context is set by specifying one of the following, mutually exclusive, options:
  - DEFAULT_CONTEXT
  - NO_CONTEXT
  - PASS_ALL_CONTEXT
  - PASS_IDENTITY_CONTEXT
  - SET_ALL_CONTEXT
  - SET_IDENTITY_CONTEXT

  If none of the previous options is specified, WebSphere MQ applies its own default, which is DEFAULT_CONTEXT.

OPEN QUEUE statement

Function Opens a previously defined queue.

Syntax

```
OPEN QUEUE {%variable | entname |
  external_qmanager:external_queue}
  [[DYNAMICQNAME=%variable | literal]
  [MODEL=%variable | literal]]
  [[option...] [?%variable...]]
```

Where

- Queue to operate on is specified as:

Queue as…	Specifies...
%variable	Queue name.
entname	Literal without quotation marks.
external_qmanager:external_queue	External name of a queue manager that contains the remote queue to process and the external name of the remote queue.

- DYNAMICQNAME keyword value is the internal Model 204 name from a DEFINE QUEUE command of a preallocated WebSphere MQ model queue that was defined with the WebSphere MQ MQQDT_PERMANENT_DYNAMIC or MQQDT_TEMPORARY_DYNAMIC attribute.

  You can specify a DYNAMICQNAME value in one of the following ways:

  – Give the fully qualified name. It will be placed in the DYNAMICQNAME field of the WebSphere MQ MQOD structure.

  – Specify a prefix of less than 33 characters for the name, followed by an asterisk (*). The queue manager generates the rest of the name.

    For example, you might want each user to use a certain prefix, or you might want to give a special security classification to queues with a certain prefix in their name.

  – Let the queue manager generate the full name. To use this method, specify an asterisk (*) in the first character position of the DYNAMIC-QNAME field.

  You must specify the DYNAMICQNAME and MODEL keywords together, otherwise, MQ/204 issues the following message:

  ```
  M204.2516: MQ/204 OPTION ERROR: options REQUIRE options
  ```

- MODEL keyword value is the internal, Model 204 name of the model queue.

When you specify the MODEL keyword you must also specify the DYNAMICQNAME keyword, otherwise, MQ/204 issues the following message:

```
M204.2516: MQ/204 OPTION ERROR: options REQUIRE options
```

- *option* is one or more of the following options:
  - ACCEPT_TRUNCATED_MSG or NO_ACCEPT_TRUNCATED_MSG
  - ALL_MSGS_AVAILABLE or NOT_ALL_MSGS_AVAILABLE (default)
  - ALLOW_PASS_ALL_CONTEXT
  - ALLOW _PASS_IDENTITY_CONTEXT
  - ALLOW_SET_ALL_CONTEXT
  - ALLOW_SET_IDENTITY_ONTEXT
  - BROWSE
  - CONTEXT
  - EXPIRY
  - FAIL_IF_QUIESCING or NO_FAIL_IF_QUIESCING
  - FORMAT
  - INPUT_AS_Q_DEF (default for local queues) or INPUT_EXCLUSIVE or INPUT_SHARED
  - LOGICAL_ORDER or NOT_LOGICAL_ORDER (default)
  - MSGTYPE
  - OUTPUT (default for remote queues)
  - PASS_USER_CONTEXT
  - PERSISTENT or NOT_PERSISTENT or PERSISTENCE_AS_Q_DEF
  - PRIORITY or PRIORITY_AS_Q_DEF
  - REPLY_QMGR
  - REPLY_QUEUE
  - REPORT
  - SAVE_ALL_CONTEXT
  - WAIT or NO_WAIT
  - WAIT_TIME
- *?%variable* specifies option(s) of the OPEN QUEUE statement to compile at evaluation time.

Usage

The queue manager for the queue must already be started. If the queue is the first queue that the user is opening for a queue manager, then MQ/204 attempts to connect the user to the queue manager prior to opening the queue. This statement performs a WebSphere MQ MQOPEN operation, and manages connections to the queue managers as needed using MQCONN and MQDISC.

Simultaneous input and output are supported, as are browse and output.

MQ/204 OPEN statement and QUEUE keyword

MQ/204 has an upward compatibility issue that is caused by supporting the keyword QUEUE on OPEN statements. If you have a Model 204 file or group named QUEUE and you use the User Language OPEN statement with it, this might conflict with the OPEN QUEUE statements, because the FILE and GROUP keywords are optional on OPEN statements.

To address this issue, OPEN statement processing identifies the case of a file or group named QUEUE. If OPEN is followed by the keyword QUEUE, the system looks ahead for a message token following the word QUEUE.

If…	The statement is assumed to be…
Message token is found	MQ/204 OPEN QUEUE statement.
No message token is found after the word QUEUE	OPEN for a file or group named QUEUE.

Note: If the MQ/204 feature is not linked in, OPEN statement parsing is unaffected.

Specifying a local dynamic queue name

The name of the model queue in the WebSphere MQ ObjDesc parameter is replaced with the name of the local dynamic queue created when the call has completed.

When a local dynamic queue is created via MQOPEN, Model 204 calls the routines to set up control blocks as if the user had entered the DEFINE QUEUE *dynamic-queue* command and is updated with the fully resolved name in the instances where the local dynamic queue name ended with an asterisk.

When you issue an OPEN QUEUE statement that did not create the local dynamic queue by referring to a model queue, a partially qualified name prefix cannot compile.

Opening a remote queue

You can open a remote queue with only the remote queue, default OUTPUT option set. All other options are invalid.

Return codes and messages

$STATUS return codes

The following $STATUS codes reflect the introduction of the RFH2 header.

$STATUS value	Meaning
47	RFH2 EXPECTED BUT NOT RECEIVED
48	RFH2 IMAGE SPECIFIED, BUT TOO SMALL
49	ERROR MOVING DATA TO IMAGE
50	SIZE ERROR WITH MULTIPLE IMAGES ON PUT(1)
51	GCORE ERROR SAVING DATA WITH PUT(1) & RFH2
52	BUFLEN SPECIFIED TOO SMALL FOR RFH2 header

$STATUSD return codes

On all $STATUS return codes that indicate a failure of a WebSphere MQ API level call, the value of $STATUSD is set to the WebSphere MQ reason code. The values and meanings of the reason codes are described in applicable IBM WebSphere MQ documents. For reason codes, see *WebSphere MQ for z/OS Messages and Codes V5.3.1*, Appendix A.

To find out which option had a problem, check $STATUSD to determine which option, as listed in Table 4-6.

Table 4-6. Identifying invalid options by $STATUSD value

$STATUSD value	Indicates invalid value for
1	MSGTYPE
2	PRIORITY
3	REPLY_QUEUE
4	MSGID
5	CORRELID
6	EXPIRY
7	FEEDBACK
8	FORMAT
9	WAIT_TIME
10	CONTEXT
11	PUTAPPLTYPE

Table 4-6. Identifying invalid options by $STATUSD value (continued)

$STATUSD value	Indicates invalid value for
12	USERIDENTIFIER
13	ACCOUNTINGTOKEN
14	APPLIDENTITYDATA
15	PUTAPPLNAME
16	PUTDATE
17	PUTTIME
18	APPLORIGINDATA
19	MAXLEN
20	POSITION
21	BUFLEN
22	DESCRIPTOR
23	REPORT
24	MSGLEN
25	REPLY_QMGR
26	DYNAMICQNAME
27	MODEL
28	GROUPID
29	GSTATUS
30	Options that save context require queue to be open with an input option
31	Options that allow and pass context require queue to be open for output
32	APPLORIGINDATA, PUTAPPLNAME, PUTAPPLTYPE, PUTDATE, and PUTTIME options that require the SET_ALL_CONTEXT option
33	USERIDENTIFIER, ACCOUNTINGTOKEN, and APPLIDENTITYDATA options that require SET_IDENTITY_CONTEXT or SET_ALL_CONTEXT
34	DYNAMICQNAME and MODEL options that must be specified together
35	MSGTOKEN
36	SEQUENCE

For all other values of $STATUS, $STATUSD is set to 0.

5

Dictionary/204

In this chapter

- Dictionary/204 overview
- FILEMGMT screen layout changes
- Field Name List screen
- Field Attributes screen
- File Sizing, Initialize and Delete screens
- Dictionary Metadata file changes

Dictionary/204 overview

This chapter contains installation and migration information pertinent to Model 204 V7R1.0 Dictionary/204.

Model 204 V7R1.0 has several features that required enhancements to the Model 204 Dictionary. This section describes the enhancements to Dictionary/204 and FILEMGMT to support the following new functionality: Table X, to contain extension records of base records stored in Table B

Setting XSIZE

To create a file with a Table X, you must set the XSIZE parameter to greater than zero. The default setting for XSIZE is zero, so that a file created in V7R1.0, by default, is created as a V6R1.0 file. If you specify a nonzero value for XSIZE, then the file is created as a V7R1.0 file.

Several screens in FILEMGMT were modified for this new functionality. Each of these screens is described in this section.

- User-defined date/time stamp support
- Optimized field extraction

New Model 204 file structure, Table X that contains the extension records of base records that are stored in Table B

RENAME FIELD prohibited for DTSFN

Date/time stamp functionality prohibits a RENAME FIELD command for the date/time stamp field name (DTSFN) field when FOPT=X’10’ is set.

These new features affect the following FILEMGMT functionality:

- Create and recreate screens
- Field Name List screen
- Field Definition screen
- File Sizing, Initialize and Delete screens
- File Parameter Reset screen
- File Increase and Decrease screens
- Dictionary metadata file changes
- FILEMGMT-generated Model 204 commands

FILEMGMT screen layout changes

Create and Recreate screens

The screens, Create and Recreate, were changed as follows:

- Add new Update Date and Time Stamp support, which is indicated by a X’10’ in the File Option (FOPT) parameter. A file with FOPT=X’10’ setting:
  - Must be an LPU file
  - May be set for a file only if the date/time stamp field specified in the DTSFN= system parameter has already been defined to the file
- Add Table X support for:
  - File create 7.1 for files with XSIZE greater than 0
  - Auto increase
  - Files with Table X have only the file organizations of entry order or unordered
- Add optimized field extract

The altered screens are shown in the following figures. The changed or additional text is highlighted in **bold**.

```
FILEMGMT               File Definition - Organization     PAGE 1 OF 3

Filename: VEHICLES FICREATE: 7.1 LANGUAGE:          Active/Inactive I
FILE ORGANIZATION    Optimized Field Extract: N       FILEORG  X' 0E '
   Entry:__  Sorted:_  Hash:X  Unordered:_     Reuse Record Numbers:_
   Sort/Hash Key:              Req Evry Rec:X  Enable Skewing:_
Table B Auto Increase Pages: 0
Table D Auto Increase Pages: 0
Table X Auto Increase Pages:
File Models:    None X  1nf _  Numeric Validation_    FILEMODL X' 00 '
FILE OPTIONS                                          FOPT     X' 00 '
   Disable Lock Pending Updates:   _  Only FM Defines New Fields:   _
   No Procedure Statement Numbers: _  No Procedure Statement Labels:_
   Enable Date Time Stamp:         N
FILE RECOVERY OPTIONS                                 FRCVOPT  X' 00 '
   Disable Transaction Back out:   _  No Rollforward (RF):          _
   No RF Logging:                  _  No Checkpoint (CK) Logging:   _
   No Updates Without RF Logging:  _  No Updates Without CK Logging:_
   All Updates Applied During RF:  _  No Discontinuities Allowed:   _
   Format File During CREATE:      Y  Verify Disk Updates:          N
===>
1=HELp   2=         3=QUIt    4=FIElds    5=WIDen    6=COMmands
7=       8=FORward  9=SIZe   10=         11=        12=END
```

Field Name List screen

The Field Name List screen was changed to support the following:

If the Update Date and Time Stamp file option is set to Y on the File Definition - Organization screen, then when you navigate to the Field Name List screen, the value from the Date Time Stamp Field Name (DTSFN) parameter is placed as the first field on the Field Name List screen.

- The DTSFN field name is required for all files that use the date/time stamp feature.
- The default value for DTSFN is left to each site. The comment in the following screen, MY DATE TIME STAMP FIELD NAME CONTAINING SPACES, illustrates where to place your chosen name.

```
FILEMGMT                    Field Name List                    Line 1 of 12

   Enter and select with an 'X' the fields you want to DEFINE to the file
Filename: VEHICLES
  |      Fieldname                                                      Cmd
  ------------------------------------------------------------------------
X MY DATE TIME STAMP FIELD NAME CONTAINING SPACES
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________
_ ______________________________________________________________________

===>

1=HELp       2=DOCument   3=QUIt        4=FILe      5=            6=COMmands
7=           8=FORward    9=RECords   10=          11=ATTributes 12=END
```

Field Attributes screen

The Field Attributes screen has changed to support the following.

The date/time stamp field:

- Must be STRING
- Cannot be defined as INVISIBLE
- Cannot be a multiply occurring field
- Can be pre-allocated

  CCA recommends ORDERED CHARACTER

```
FILEMGMT                          Field Attributes

FILENAME: VEHICLES
FIELDNAME: MY DATE TIME STAMP FIELD NAME CONTAINING SPACES

Data Type NON-PREALLOCATED  Average Length                          Security
STRING       REPEATABLE      IN BYTES 17__                          LVL: 0__

UPD IN PLACE                              MIN. OCCURS 1__      NO. OF UNIQUE
                                          MAX. OCCURS 1__      VALUES: _________
NON-KEY       NON-ORDERED
                                          ===== ACCESS/TARGET Descriptors =====
              NON-NUMERIC-RANGE           Picture       Format  FIXED
                                          X(17)_______  NO MIXED CASE




===>

1=HELp        2=            3=QUIt    4=RECords    5=COPy       6=COMmands
7=PREv value  8=NEXt value  9=SIZe   10=PRVfield  11=NXTfield  12=END
```

File Sizing, Initialize and Delete screens

These screens are all basically the same screen with different functionality and have been updated to support the Table X feature.

File Size Definition Screen

```
FILEMGMT               File Size Definition Screen

FILENAME: VEHICLES                                  Inactive
FICREATE: 7.1
FILEORG:  Hashed Order                                 PAGESZ: 6184
ASIZE: 3      BSIZE: 57      DSIZE: 4       XSIZE: 0          PROCS
------------  ------------   ------------   ------------   ----------
ASTRPPG: 511  BRECPPG: 233   DRESERVE: 15 % XRECPPG:   0   CUR:  0
ATRPG:   1    BRESERVE: 29   PDSIZE:    1   XRESERVE:  0   EXP:  0
FVFPG:   1    RECS:  11000   PDSTRPPG:128   RECRDOPT:  0   AVPG: 1
MVFPG:   1                   DPGSRES:  2
CSIZE:   1                ESIZE:   0
FREESIZE:                 Total Space Pages:         73

   DDname    Dataset Name   ( 1 of 3 )            Device    Trks/Blks
  USER5XXX  MMUUSER5.USER5XXX                     3390      10
  ________  ____________________________________  ________  __________
  ________  ____________________________________  ________  __________
===>

1=HELp   2=RECords   3=QUIt    4=FIElds    5=           6=COMmands
7=       8=FORward   9=SIZe   10=FILe     11=          12=END
```

File Parameter Reset screen

The File Parameter Reset screen has been changed to support the following:

- Table X automatic increase parameter, XAUTOINC
- Enable Date Time Stamp option setting

```
FILEMGMT                File Parameter Reset Screen        Page 1 of 2
Filename: METADATA     FICREATE: 7.1  LANGUAGE: US         Inactive
FILE MODEL OPTIONS                                     FILEMODL X' 00 '
  None:X    1nf:_    Numeric Validation:_

TABLE B AND D AND X AUTOMATIC INCREASE OPTIONS:
Table B Auto Increase: Pages: __
Table D Auto Increase: Pages: __
Table X Auto Increase: Pages: __
BRSV: 200    DRSV: 15 % DPGSRES: 2      XRSV:      BREUSE:
FILE OPTIONS:                                        FOPT     X' 00 '
  Disable Lock Pending Updates: _  Only FM Defines New Fields:     _
  No Procedure Statement Labels:_  No Procedure Statement Numbers: _
  Enable Date Time Stamp:       N
FILE RECOVERY OPTIONS                                FRCVOPT  X' 00 '
  Disable Transaction Backout:  _  No Rollforward (RF):            _
  No RF Logging:                _  No Checkpoint (CK) Logging:     _
  No Updates Without RF Logging:_  No Updates Without CK Logging:  _
  All Updates Applied During RF:_  No Discontinuities Allowed:     _
===>                                                               <
FIM314 You have selected Datetime Stamp, hit PF11 to define attributes
1=HELp   2=         3=QUIt     4=FIELDs  5=             6=COMmands
7=       8=FORward  9=        10=       11=ATTributes  12=END
```

The Enable Update Date Time Stamp entry on this screen will default to N.

File Increase/Decrease screen

```
FILEMGMT                 File Increase/Decrease

Filename:   VEHICLES                                    Inactive
Fileorg:    Hashed Order
TOTAL SPACE     Pages:  73
                        New Sizes,                New Sizes
BSIZE:,    57           57      DSIZE:  4         4
BHIGHPG:                        DPGSUSED:

ESIZE:     0            0       XSIZE:
EHIGHPG:                        XHIGHPG:
FREESIZE:  5            5
  DDname               Dataset Name ( 1 OF 5 )    Device    Trks/Blks
VEHICLES  PSEDMNT.VEHICLES_____________________  ________  _________
________  ____________________________________  ________  _________
________  ____________________________________  ________  _________
________  ____________________________________  ________  _________
________  ____________________________________  ________  _________
===>

1=HELp     2=         3=QUIt     4=         5=         6=COMmands
7=         8=FORward  9=        10=        11=        12=END
```

Dictionary Metadata file changes

New fields were added to Metadata to support Table X information:

- XAUTOINC
- XHIGHPG
- XRECORDS
- XRECPPG
- XRESERVE
- XSIZE

A new field was added to Metadata to support the Date/time Stamp feature:

- Date/time Stamp Field Name (DTSFN)

FILEMGMT generated Model 204 commands

The following is an example of the Model 204 commands that are generated after the implementation of the Dictionary/204 enhancements.

```
CREATE MARA10
PARAMETER ADDLVL=0
PARAMETER ASTRPPG=383
PARAMETER ATRPG=1
PARAMETER BAUTOINC=5
PARAMETER BRECPPG=36
PARAMETER BRESERVE=186
PARAMETER BSIZE=496
PARAMETER CSIZE=1
PARAMETER DAUTOINC=5
PARAMETER DRESERVE=15
PARAMETER DSIZE=304
PARAMETER ESIZE=15020
PARAMETER XAUTOINC=5
PARAMETER XRECPPG=10
PARAMETER XRESERVE=150
PARAMETER XSIZE=100
PARAMETER FOPT=X'01'
PARAMETER FILEORG=X'00'
PARAMETER FRCVOPT=X'C0'
PARAMETER FVFPG=1
PARAMETER IVERIFY=X'00'
PARAMETER MVFPG=1
PARAMETER PDSIZE=1
PARAMETER PDSTRPPG=128
PARAMETER PRCLDEF=0
PARAMETER READLVL=0
PARAMETER SELLVL=0
PARAMETER UPDTLVL=0
```

```
PARAMETER BREUSE=0
PARAMETER DPGSRES=2
PARAMETER FILEMODL=X'00'
PARAMETER LANGFILE=US
PARAMETER OPENCTL X'80'
END

* Close all files in case OPEN fails
CLOSE ALL
OPEN FILE MARA10
IN FILE MARA10 INITIALIZE
RESET  PRIVDEF X'BFFF'
DEFINE FIELD 'RUNNER ID' (NKEY NNR NFRV NDEF UP LVL 0 ORD CHAR NUNIQ LRES 15 -
 NRES 15 SPLT 50 IMMED 1 REPT  NCOD VIS OCC 1 LEN 5 PAD X'00' )
DEFINE FIELD 'LASTNAME' (NKEY NNR NFRV DEF UP LVL 0 ORD CHAR NUNIQ LRES 15 -
 NRES 15 SPLT 50 IMMED 1 REPT   NCOD VIS)
DEFINE FIELD 'FIRSTNAME' (NKEY NNR NFRV UP LVL 0 NORD REPT   NCOD VIS)
DEFINE FIELD 'SEX' (KEY NNR FRV DEF UP LVL 0 NORD REPT   FV NCOD VIS OCC 1 LEN-
 1 PAD X'00')
DEFINE FIELD 'BIRTHDATE' (NKEY NNR NFRV DEF UP LVL 0 ORD NUM NUNIQ LRES 15 -
 NRES 15 SPLT 50 IMMED 1 REPT   FV COD VIS BIN OCC 1)
DEFINE FIELD 'ADDRESS' (NKEY NNR NFRV UP LVL 0 NORD REPT   NCOD VIS)
DEFINE FIELD 'CITY' (NKEY NNR NFRV DEF UP LVL 0 ORD CHAR NUNIQ LRES 15 NRES 15-
 SPLT 50 IMMED 1 REPT  NCOD VIS)
DEFINE FIELD 'STATE' (NKEY NNR NFRV DEF UP LVL 0 ORD CHAR NUNIQ LRES 15 NRES 1-
5 SPLT 50 IMMED 1 REPT   NCOD VIS OCC 1 LEN 2 PAD X'00')
DEFINE FIELD 'ZIP' (NKEY NNR NFRV UP LVL 0 NORD REPT   NCOD VIS)
DEFINE FIELD 'AGE' (NKEY NNR NFRV DEF UP LVL 0 ORD NUM NUNIQ LRES 15 NRES 15 -
 SPLT 50 IMMED 1 REPT   NCOD VIS)
DEFINE FIELD 'MIDDLE INIT' (NKEY NNR NFRV UP LVL 0 NORD REPT   NCOD VIS)
DEFINE FIELD 'COUNTRY' (NKEY NNR NFRV DEF UP LVL 0 ORD CHAR NUNIQ LRES 15 NRES-
 15 SPLT 50 IMMED 1 REPT   FV COD VIS)
DEFINE FIELD 'PHONE NUMBER' (NKEY NNR NFRV UP LVL 0 NORD REPT   NCOD VIS)
DEFINE FIELD 'ENTRY FEE' (NKEY NNR NFRV UP LVL 0 NORD REPT   NCOD VIS)
DEFINE FIELD 'OCCUPATION' (NKEY NNR NFRV UP LVL 0 NORD REPT   FV COD VIS)
DEFINE FIELD 'FINISH TIME' (NKEY NNR NFRV DEF UP LVL 0 ORD CHAR NUNIQ LRES 15 -
 NRES 15 SPLT 50 IMMED 1 REPT   NCOD VIS)
DEFINE FIELD 'FINISH PHOTO' (NKEY NNR NFRV UP LVL 0 NORD REPT  BLOB NCOD VIS)
CLOSE MARA10
```

6

New and Enhanced Parameters and Commands

Parameters introduced or modified after V6R1.0

- BRLIMSZ: Table B base record limit size
- CDMAXP2X: Compacted CCATEMP page allowance for constraints log
- CHKPRIV: System administrator CHECKPOINT command privileges
- COMPERR: Compiler errors reporting options
- CUSTOM: Using customized options
- DEFINCP: Default automatic Tables B, D, and X size percentage increase
- DSPOPT: Data- and hiperspace options
- DTS: Enables automated date/time stamp on file updates
- DTSFN: Name of the date/time stamp field
- ERMX: Maximum number of errors for user's session
- ERRMSGL: Number of bytes used for saved error messages
- FICREATE: File creation release number
- FILEORG: File organization

- FOPT: File options
- HASHCELL: Hash cells per buffer
- IPADDR: Telnet server IP address display
- KOMMOPT: Multiple user KOMMs
- LDKBMWNG: Number of pages in above the bar anticipatory write window
- LPDLST: User Push Down List length
- MAXINCBP: Percentage of maximum dynamic increase for Table B
- MAXSIMIO: Maximum number of simultaneous disk I/Os
- MQSUBREL: MQ subtask release control
- NLRUQG: Number of above the bar DKBM least recently used (LRU) queues
- NUMBUFG: Number of buffers allocated above the bar
- PRIORITY: Priority class
- RECRDOPT: Record options
- SCHDOPT: Scheduler operation and accounting
- SMTPCADD: SMTP client address
- SMTPCPRT: SMTP client port
- SMTPSADD: SMTP server address
- SMTPSPRT: SMTP server port
- SNAPFAIL: Number of CCASNAP failures
- SNAPFLIM: Snap failure limit
- SYSOPT2: System options
- UCPUSLIC: CPU time slice allotment for an individual user
- UIOSLIC: I/O time slice allotment for individual user
- USLCMAX: Number of stop-loop-checks before CSLICE is invoked
- USLCWAIT: Sleep time at minimum dispatching priority
- XAUTOINC: Number of unused pages maintained in Table X by automatic increase
- XHIGHPG: Table X highest active page
- XQLEN: Table X queue length

- XRECPPG: Table X extension record slots per page
- XRESERVE: Table X reserve space per page
- XREUSE: Free space required to reuse Table X page
- XREUSED: Extension slots reused in Table X
- XSIZE: Pages in Table X

Commands introduced or modified after V6R1.0

- CHECKPOINT: Requesting and handling a checkpoint
- COMPACTB: Invoking the data compactor
- COMPACTE: Table E compactor to reduce fragmentation
- DECREASE: Decreasing the file size
- DEFINE DATASET: Large dataset characteristics
- DELETE: Field
- FILELOAD: Starting a File Load program (single-step)
- FLOD: Starting a File Load program (multistep)
- INCREASE: Increasing the file size
- MONITOR: An Online system
- MONITOR SIZE: Tracking journal and checkpoint streams
- MSGCTL: Setting message output
- PRIORITY: Assigning a priority class
- REGENERATE: Recovering a file in two passes
- RENAME FIELD: Renaming a field
- SWITCH STREAM: Change stream to next member
- TABLEB: Determining usage of Table B
- TABLEBX: Determining use of Tables B and X
- TABLEE: Determining Table E use
- TABLEX: Determining Table X use

Individual parameters

BRLIMSZ: Table B base record limit size

Status New in V7R1.0

Class TABLES

Default 0

Setting View-only

Meaning Displays the base record size limit that was set in files with RECRDOPT=1.

CDMAXP2X: Compacted CCATEMP page allowance for constraints log

Status New in V7R1.0, CDMINP2X cannot exceed the value of CDMAXP2X.

Class SYSTEM

Default 8

Setting On User 0's parameter line

Meaning The CDMAXP2X parameter indicates the maximum number—in powers of 2—of 6K pages allocated in virtual storage that will be used for the TBO constraints log, and in CCATEMP, if CDMAXP2X is greater than CDMINP2X. These pages are permanently allocated during initialization. Pages are compacted till their number exceeds CDMAXP2X.

When a constraints page becomes almost full, it is split into two pages. When the number of pages exceeds 2**CDMAXP2X, the splitting process stops and records grow into overflow pages that never split, but may be merged when they become too empty. Overflow pages are not compacted, because they do not have basic records and thus do not participate in hashing. The only way to find data on overflow pages is to look through all cells.

For example:

- CDMAXP2X=0 sets the maximum number of pages to compact to one.
- CDMAXP2X=8 sets the maximum number of pages to compact to 256.

The minimum value is zero and maximum value is 16, which corresponds to 64K pages. The parameter is not resettable.

However, during initialization, if the value of CDMAXP2X is less than the value of CDMINP2X, then CDMAXP2X is automatically reset to the larger value of CDMINP2X. The following message is also issued:

```
M204.2777: CDMAXP2X PARAMETER RESET TO LARGER CDMINP2X
```

Increasing the number of compacted pages may save CCATEMP pages, however, compacting more pages may consume more CPU time. CCA recommends setting the value of CDMAXP2X between 7 and 9 and testing your results.

CHKPRIV: System administrator CHECKPOINT command privileges

Status New in V7R1.0

Purpose Allows any system administrator to issue specified CHECKPOINT commands based on the setting of the CHKPRIV parameter.

Class USER

Default X’01’

Setting In CCAIN User 0 parameters

Meaning Any system administrator can issue one or more forms of the CHECKPOINT command depending on the following settings.

Setting	System administrator can do…
X’01’	TRAN and/or SUBTRAN
X’02’	MESSAGE
X’04’	ABORT
X’08’	EXTENDED QUIESCE

COMPERR: Compiler errors reporting options

Status New in V7R1.0

Purpose To limit the number of error messages generated during compilation that are displayed and to highlight the errors for you.

Class USER

Default X’04’

Setting In CCAIN User 0 parameters

Meaning You can choose the volume of compiler errors you want returned to you and how you want them displayed, so you can locate the problem more easily.

Usage You can sum the following, possible settings:

Setting	Meaning
X'01'	Point to last token read on a compile error in the following format: `>> last token read <<`
X'02'	Underline last token read on a compile error in the following format: last token read
X'04'	All multiple compile errors to be issued per line (default behavior)

CUSTOM: Using customized options

Status Changed in V7R1.0

Class SYSTEM

Default None

Setting On User 0's parameter line or reset by the system manager

Meaning The CUSTOM parameter enables special CCA-written modifications to standard execution.

Using CUSTOM=(9)

The CUSTOM=(9) option lets you suppress all output from all forms of the PRIORITY command.

Using CUSTOM=(7)

The CUSTOM=(7) is obsolete in V7R1.0.

DEFINCP: Default automatic Tables B, D, and X size percentage increase

Status New in V7R1.0

Class SYSTEM

Default 0

Setting Set in JCL on EXEC statement as a PARM argument, in CCAIN User 0 parameters, or by system manager

Meaning A percentage of the current Table B size (BSIZE), Table D (DSIZE), and Table X (XSIZE) by which each table can be increased dynamically as a result of automatic increase.

Usage DEFINCP helps to avoid shutting down a file to increase its Tables B, D, and/or X. The amount of the increase for Table B is also limited by the current number of segments in the file, if MAXINCBP is not set.

Example If a file has 100,000 Table B pages and DEFINCP is set to 10, and Table B fills up, the size of Table B will increase to 110,000; then 121,000; then 133,100 pages and so on. The amount of increase is subject to FREESIZE and MAXINCBP restrictions.

DSPOPT: Data- and hiperspace options

Status X’80’ option new in V7R1.0

Class SYSTEM

Default X’00’

Setting On User 0’s parameter line

Meaning The bits have the following meanings:

Settings	Meaning
X’00’	APSY subsystem saved precompiled procedures are not kept in CCATEMP, if they have been moved to the APSYPAGE storage area.
X’01’	Use page-oriented movement for servers, which means use virtual storage 4K-pages instead of byte-oriented movement. Page-align servers and use the MVPG hardware instruction to do CCASERVR I/O. Using page-aligned CCASERVRs lets Model 204 use a page-oriented storage allocation algorithm that probably requires a significantly smaller working set size than using non-page-aligned CCASERVRs. Not supported for a z/VM operating system.
X’02’	Put servers in a hiperspace instead of a dataspace.
X'20'	Put APSY subsystem precompiled procedures in hiperspace instead of a dataspace. Hiperspace pages can be in either real or expanded storage. This option is valid on only z/OS and requires that Model 204 be APF-authorized.

Settings	Meaning
X'40'	Put APSY subsystem precompiled procedures in cache hiperspace instead of a scrolling hiperspace. Cache hiperspace pages can reside in only expanded storage, so they are sometimes called ESO (for Expanded Storage Only) hiperspaces. Also, cache hiperspace pages never result in operating system paging. Pages stolen by the operating system are simply discarded and Model 204 is responsible for replacing the pages from the copy of the precompiled procedures in CCATEMP. This option is valid on only z/OS and requires that Model 204 be APF-authorized.
X'80'	Keep all APSY subsystem saved precompiled procedures in CCATEMP. Consider including DSPOPT=X'80' only when TEMPPAGE=0.

DTS: Enables automated date/time stamp on file updates

Status New in V7R1.0

Class USER

Default 1

Setting Reset by system manager

Meaning Enables date/time stamp processing when set to one. A system manager can reset DTS to zero to disable automatic date/time stamps. Turning off the date/time stamp feature does not imply that you cannot update any field, including the date/time stamp field; it simply stops automatic updates to the date/time stamp field.

Values	Behavior
1	For date/time stamp files, transactions participate in DTS updating. The date/time stamp field of an updated record is automatically updated at the beginning of COMMIT processing. This is the default setting for the parameter.
0	Transactions do not participate in DTS updating. When a transaction does not participate in DTS processing, the date/time stamp field will not be updated on records that are themselves updated.

The DTS parameter may be reset only by a user with system manager privileges. In addition, the user must have file manager privileges for any DTS-enabled files opened, either before or after DTS is reset to 0. Turning off the DTS parameter affects updates only to date/time stamp files, so no special restrictions were placed on the file with no date/time stamp.

Usage When DTS=0, then your user thread does no date/time stamp processing on any file. To turn DTS processing off for a file, you must stop the file and turn off the FOPT=X'10' bit. This suspends date/time stamp processing for this file for all users.

Currently, there is no feature that can suspend date/time processing for some files for some users.

Furthermore, DTS is not supported under Parallel Query Option. If you attempt a remote open of a data/time stamp file (FOPT=X'10), the following message is issued:

```
M204.1977: %F MAY NOT BE ACCESSED REMOTELY
```

DTSFN: Name of the date/time stamp field

Status New in V7R1.0

Class SYSTEM

Default None

Setting On User 0's parameter line

Meaning The name of the date/time stamp field for the current run of Model 204 for all date/time stamp files involved in the run.

If the field name contains spaces, you must use the character string format (C' '). See the "Requirements for parameter values" section in the *Model 204 Parameter and Command Reference*.

ERMX: Maximum number of errors for user's session

Status New processing change in V7R1.0

Class USER

Default 30

Setting By any user

Meaning The maximum number of counting errors before either user restart or compilation failure.

- If ERMX is exceeded due to failed commands within a procedure, the user is restarted softly.

- If ERMX is exceeded due to User Language compilation errors, compilation stops and the user is returned to command level.
- If ERMX is exceeded due to User Language evaluation (run time) errors, the request is terminated and the user is restarted softly.

The valid range for setting a limit is 0 to 65535.

Setting a value of -1 or 65535 ignores counting errors and processing continues.

ERRMSGL: Number of bytes used for saved error messages

Status New in V7R1.0

Class SYSTEM

Default 80

Setting On User 0's parameter line

Meaning The ERRMSGL parameter indicates the number of bytes to use for saved error messages—those messages returned by $ERRMSG and $FSTERR. The minimum value of ERRMSGL is 80 and the maximum is 256.

The actual length of saved messages is one less than the setting of ERRMSGL, so that the default ERRMSGL of 80 can save up to 79 bytes of error messages.

In Model 204 V6R1.0 and earlier, only 79 bytes of error messages could ever be saved, so it was as if the setting of ERRMSGL was 80.

FICREATE: File creation release number

Status Settings 7 and 8 are new in V7R1.0

Class FPARMS

Default None

Setting View-only

Meaning The release of Model 204 in which the file was created

Valid settings of FICREATE are:

Setting	File created in…
0	Release 7.1 or an earlier release of Model 204.

Setting	File created in…
1	Release 8.0 of Model 204.
2	Release 8.1 of Model 204.
3	Release 9.0 (or later) of Model 204.
4	Version 2 Release 2 (or later) of Model 204, and also has a nonzero setting of the FILEMODL parameter.
5	Version 3 Release 2 (or later) of Model 204, and also has a nonnull setting of the LANGFILE parameter.
6	Version 6 Release 1 (or later) of Model 204.
7	If you create a file with non-zero XSIZE, but no ESIZE, then FICREATE is set at 7. Files with FICREATE=7 cannot be opened by Model 204 V6R1.0 or earlier.
8	If you create a file with non-zero ESIZE, with or without a non-zero XSIZE, then FICREATE set at 8 specifies Model 204 V7R1.0.

FILEORG: File organization

Status Setting X’80’ New in V7R1.0

Class FPARMS

Default 0

Setting During file creation

Meaning The file (Table B) organization

Valid settings of FILEORG are:

Setting	Meaning
X’80’	All nonpreallocated (non-OCCURS) fields are preceded by a field-value length byte. With a length byte on every field, even FLOAT, CODED or BINARY fields, several instructions and one IF test are eliminated from the internal field scan loop. Having a length byte also allows some simple compression of BINARY, CODED, and FLOAT values.
X‘40'	File skewing enabled. (This option is obsolete and no longer recommended by CCA. See the *Model 204 File Managers Guide* for a discussion of its history.)
X‘20'	Unordered file.
X‘08'	Hash key file.

Setting	Meaning
X‘04'	Reuse record number (RRN). This option can be added to any other FILEORG option. If this option is set without the hash key (X‘08') or sorted (X‘01') options, the unordered option (X‘20') is automatically set.
X‘02'	Sort or hash key required in every record. (This option can be added to the X‘08' and X‘01' options.)
X‘01'	Sorted file.
X‘00'	Entry order file.

FOPT: File options

Status New in V7R1.0: FOPT=X’10’ that enables the date/time stamp feature

Class FPARMS

Default 0

Setting During file creation or reset by file manager

Meaning The following table lists the valid settings for file options, which can be summed:

Setting	Meaning
X‘80'	Prohibits statement numbers in procedures.
X‘40'	Prohibits statement labels in procedures.
X’10’	Enables the date/time stamp feature.
X‘08'	(Obsolete) Indicates append-first mode in an RDFS file. Without the X‘08' option, an RDFS file is in reuse-first mode. The FILEORG parameter contains option 4.
X‘02'	Disables lock pending updates — Model 204 automatically disables transaction back out by turning on the X‘08' bit of the FRCVOPT parameter. The X‘02' bit of the FOPT parameter and the X‘08' bit of the FRCVOPT parameter for a file must be turned off to activate the transaction back out facility.
X‘01'	Permits only a file manager to define new field names, not other users.
X‘00'	Allows labels and statement numbers in the same file.

Options X‘80' and X‘40' can be used to avoid the mixture of statement numbers and labels in procedures on old files and/or to ensure that any procedure development on new files uses statement labels exclusively.

Option X’10’ can be set only *after* the date/time stamp field has been defined in the file. You cannot set X’10’ when you create the file, because the date/time stamp field has not yet been created. The X’02’ and X’10’ options are mutually exclusive.

A file manager can set these file options during file creation or via the RESET command. Note that the options are file-specific. Thus, if a procedure from one file includes a procedure from another file, the use of labels or numbers within each procedure is governed by the FOPT option assigned to the file where they reside.

The enforcement of these options occurs at compile time and is in effect only for procedures compiled from a Model 204 file. These options are not enforced if a procedure is compiled as a temporary procedure.

Note: When a file is opened, bits that are not currently defined by CCA for FOPT are reset. This bit resetting is designed to permit the possible use of these bits by features to be introduced in future Model 204 releases. You cannot use the RESET command to set bits that are currently undefined by CCA.

If an application makes use of any of the undefined bits of the FOPT parameter, it can produce unexpected results.

HASHCELL: Hash cells per buffer

Status New in V7R1.0

Class SYSTEM

Default 3

Setting On User 0’s parameter line

Meaning The HASHCELL parameter indicates the number of hash cells to allocate for each buffer in the buffer pool. Hash cells are used to locate database pages in the buffer pool based on file, table, and page number.

The default (and minimum) value for HASHCELL is 3 and the maximum value is 7. CCA recommends resetting the default only when running with AMPSUBS>0.

IPADDR: Telnet server IP address display

Status: New in V7R1.0

Class SYSTEM

Default Displays the IP address of the z/OS Telnet server, if available

Setting View-only

Meaning z/OS systems have a TCP-based Telnet server to accept TN3270 connections. When you, a PC user, connect to z/OS via TCP/IP, the Telnet server creates both a TCP connection between the PC and the Telnet server and an LU-2 session with VTAM.

On the PC Telnet client you see the VTAM menu as though you are a local VTAM terminal. For example, you can connect to Model 204 through the VTAM menu. Model 204 treats your session as an IODEV=7, even though you are connected to VTAM via TCP/IP. Model 204 applications can run without modification in this configuration, unaware there is a TCP connection involved.

However, you might want to know that you are actually connected via TCP/IP for tracking purposes or performance analysis. To find this out, issue the VIEW IPADDR command to display the IP address, if it is available.

KOMMOPT: Multiple user KOMMs

Status Obsolete as of Model 204 V7R1.0.

In Model 204 V7R1.0 and later all multi-user jobs have a KOMM for each user (or multiple KOMMs), which is the equivalent of the obsolete KOMMOPT=1 setting. This change should decrease your server size requirements.

For more information about KOMMs and other Model 204 data structures, refer to the *Model 204 System Manager's Guide*.

LDKBMWNG: Number of pages in above the bar anticipatory write window

Status New in V7R1.0

Class SYSTEM

Default The value of LDKBMWND, if it is set, or 3.

Setting On User 0's parameter line by system manager

Meaning An anticipatory write window that starts writes for any modified buffers that enter the window. When an anticipatory write for a modified buffer completes, the buffer is immediately reusable.

LPDLST: User Push Down List length

Status New minimum and default value in V7R1.0

Class UTABLE

Default 5000 bytes

Setting On user's parameter line or reset by UTABLE command

Meaning The length of the User Push Down List (PDL), a work area used for Model 204 program flow control.

LPDLST also includes some space for the C pattern matcher processing. Complex patterns (for example, patterns involving nested repeated patterns or nested wild cards) might overflow the LPDLST area and require you to increase the LPDLST setting to 32K.

LPDLST can range from 5000 to as large as 32760. Its recommended initial value for SQL processing is the maximum.

You can monitor LPDLST use with the PDL audit trail statistic.

If you get unrecoverable errors when running User Language requests, you might need to increase the size of LPDLST (by approximately 10-20%, depending on your environment and the applications being run) and/or the LQTBL and LSERVPD parameters.

Because check pointing is asynchronous in 31-bit operating systems, you might need to increase LPDLST in such environments.

MAXINCBP: Percentage of maximum dynamic increase for Table B

Status New in V7R1.0

Class SYSTEM

Default 0

Setting Set in JCL on EXEC statement as a PARM argument, in CCAIN User 0 parameters, or by system manager.

Meaning You can increase Table B for a file dynamically up to MAXINCBP percent without taking the file out of service.

MAXSIMIO: Maximum number of simultaneous disk I/Os

Status New in V7R1.0

Class SYSTEM

Default 32 * (NSUBTKS + NSERVS), if using IOS Branch Entry (XMEMOPT=2)

(32 * (NSUBTKS + NSERVS)) + LDKBMWND, if not using IOS Branch Entry

Setting By system manager or on User 0's line

Meaning Limits DBIDs when using IOS Branch Entry.

Usage During initialization, Model 204 calculates a default number of Disk Buffer I/O control blocks (DBIDs). If you are using IOS Branch Entry, this default number may be unnecessarily high. Use the MAXSIMIO parameter to allocate a smaller number of DBIDs.

Using a lower MAXSIMIO value, when you are also using IOS Branch Entry, limits the number of DBIDs allocated and results in storage savings. If you are not using IOS Branch Entry, an explicit MAXSIMIO setting is ignored and the value calculated by Model 204 is used instead.

MQSUBREL: MQ subtask release control

Status New in V7R1.0

Class USER

Default 0

Setting User 0's CCAIN line or reset by user

Meaning The MQ subtask is released at the end of request processing in subsystems with AUTOCOMMIT=N.

Usage You can set the MQSUBREL parameter on User 0's CCAIN line to affect all users. Or, a user can reset it to affect only a specific user. At login or a user restart the MQSUBREL parameter is returned to the value specified by the User 0 CCAIN parameter line.

When set to…	Allows you to…
0	Retain the subtask and preserve pre-V7R1.0 behavior.
1	Release the subtask to other users.

Set this parameter to 0 when you expect the same user to use MQ services soon after ending a request. Set it to 1 when you expect other users may be waiting for a free MQ subtask.

NLRUQG: Number of above the bar DKBM least recently used (LRU) queues

Status New in V7R1.0

Class SYSTEM

Default 1, also the minimum

Setting Not resettable

Meaning The number of above the bar least recently used (LRU) queues to allocate

Usage If you do not set NLRUQG, it is set to the same value as NLRUQ. Note this results in an immediate doubling of the number of LRU queues, if you have set NUMBUFG to greater than zero,

If NLRUQG is set greater than 1, then the value of LDKBMWNG is rounded up to a multiple of NLRUQG. The maximum setting for NLRUQG is 255.

A high conflict count for LRU queue locks that would appear in MONITOR MPLOCK command output indicates that you should increase NLRUQG and NLRUQ.

NUMBUFG: Number of buffers allocated above the bar

Status New in V7R1.0

Class SYSTEM

Default 0

Setting Not resettable

Meaning Allocate the number of buffers above the bar.

The maximum number of buffers must be <2**31/3 or 715,827,882, which limits buffer pools to 4.2 terabytes or less.

PRIORITY: Priority class

Status New in V7R1.0, *(cur,min,max)* option

Class USER

Default STANDARD

Setting On user’s parameter line

Meaning The user’s priority class

PRIORITY is a character string parameter. Valid settings for PRIORITY are LOW, STANDARD, HIGH, or (*cur,min,max*), where *cur* specifies the new current priority of 0-255, *min* specifies the new minimum priority of 1-253, and *max* specifies the new maximum priority of 1-253.

A user's priority class dictates a range of allowable internal priorities. In general, HIGH priority users receive service sooner than STANDARD priority users; STANDARD priority users receive service sooner than LOW priority users. For more information on priority classes, refer to the *Model 204 System Manager's Guide*.

The setting of PRIORITY is overridden if a priority is specified in the user's login password table entry.

RECRDOPT: Record options

Status New in V7R1.0

Class TABLES

Default 0

Setting At file creation, then view-only

Meaning Record options

Usage Possible settings for the RECRDOPT parameter are:

Setting	Purpose
1	Limit base record size. When RECRDOPT is set to 1, the size of base records is limited so that all record slots on a Table B page may be used to begin logical records (base record, plus extensions). The limit is based on BRECPPG and is set on the view-only parameter, BRLIMSZ. RECRDOPT may only be set to 1 in files with XSIZE>0 which are either ordered or unordered. Hashed and sorted files are not supported.

When RECRDOPT=1 is specified, the base record limit size is calculated as follows:

- For an entry order file, BRLIMSZ = (6140 / BRECPPG) - 2
- For an unordered file, BRLIMSZ = (6136 / BRECPPG) - 2

Files created in V6R1.0 or earlier must be reorganized to use the RECRDOPT parameter.

Internally, files with BRECPPG=1 or XRECPPG=1 are treated as files with a base record limit to avoid Table B or Table X page space constraints processing.

SCHDOPT: Scheduler operation and accounting

Status Setting X’10’ new in V7R1.0

Class SYSTEM

Default 0

Setting On User 0’s parameter line or reset by system manager.

Meaning Control of the main task scheduler operation and accounting.

- Settings X’01’ to X’08’ inclusive are relevant only to sites running MP/204.
- Setting X’10’ is relevant to all sites.

Valid settings of SCHDOPT, which can be summed, are:

Setting	Meaning
X‘00'	(MP/204 only) No scheduler overhead tracking, no forced offload. This is the default.
X‘01'	(MP/204 only) Enable scheduler overhead tracking.
X‘02'	(MP/204 only) Enable forced off load.
X’04’	(MP/204 only) Enable DKPRF - fast DKPR or fast logical read duplicated on next page.
X’08’	(MP/204 only) Server swaps are allowed to be performed by MP subtasks.
X’10’	CSLICE verifies that long request values are not exceeded.

SMTPCADD: SMTP client address

Status New in V7R1.0 for $SNDMAIL function

Class SYSTEM

Default 0

Setting On User 0’s parameter line or reset by system manager

Meaning SMTPCADD is the default client address, the TCP/IP domain for the operating system, for the $SNDMAIL function. You can specify SMTPCADD as:

- Domain name of up to 255 bytes

If the first character of the address is an alphabetic character (A to Z), it is automatically evaluated as a domain name.

- Internet Protocol (IP) address, IPv4 format

You can override SMTPCADD for a specific calls by entering a different value in the EMAILD parameter of the $SNDMAIL function call.

SMTPCPRT: SMTP client port

Status New in V7R1.0 for $SNDMAIL function

Class SYSTEM

Default 7700

Setting On User 0's parameter line or reset by system manager

Meaning SMTPCPRT is the default client port for the $SNDMAIL function. The value can be any value in the user range—normally above 4095 and up to 32767.

You can override SMTPCPRT for a specific call by entering a different value in the EMAILD parameter of the $SNDMAIL function call.

SMTPSADD: SMTP server address

Status New in V7R1.0 for $SNDMAIL function

Class SYSTEM

Default 0

Setting On User 0's parameter line or reset by system manager

Meaning SMTPSADD is the default server address for the $SNDMAIL function. You can specify SMTPSADD as:

- Domain name of up to 255 bytes

  If the first character of the address is an alphabetic character (A to Z), it is automatically evaluated as a domain name.

- Internet Protocol (IP) address, IPv4 format

You can override SMTPCADD for a specific calls by entering a different value in the EMAILD parameter of the $SNDMAIL function call.

SMTPSPRT: SMTP server port

Status New in V7R1.0 for $SNDMAIL function

Class SYSTEM

Default 25

Setting On User 0's parameter line or reset by system manager

Meaning SMTPSPRT is the default server port for the $SNDMAIL function. SMTPSPRT is normally set to the standard SMTP value of 25. You can override SMTPSPRT for a specific call by entering a different value in the EMAILD parameter of the $SNDMAIL function.

SNAPFAIL: Number of CCASNAP failures

Status New in V7R1.0

Class SYSTEM

Default 0

Setting View-only

Meaning Number of times a CCASNAP has failed.

Usage When the value of SNAPFAIL reaches the value of SNAPFLIM, SNAPLIM is set to SNAPID, preventing any further CCASNAP attempts.

SNAPFLIM: Snap failure limit

Status New in V7R1.0

Class SYSTEM

Default 5

Setting Set by the system manager

Meaning Number of times SNAPs can fail before CCASNAPs are deactivated by setting SNAPLIM=SNAPID. If SNAPLIM=0, unexpected program exceptions while processing a CCASNAP still causes a 4095 abend. Otherwise, each time an unexpected program exception occurs, the SNAPFAIL, view-only parameter is incremented. The maximum setting is 255.

SYSOPT2: System options

Status Hex setting X’40” new in V7R1.0

Class SYSTEM

Default 0

Setting On User 0’s parameter line or reset by system manager

Meaning The valid settings of SYSOPT2 are (options can be summed):

Hex setting	Decimal setting	Meaning
X‘80'	128	Manages the XTIOT option.
X‘40'	64	Three-byte CCATEMP pages numbers, instead of 2-byte page numbers are used in the record locking table. This requires an increase in LRETBL. CCA recommends an increase of 50 percent. If SYSOPT2 is set to X’40’, the record locking table is placed above the bar.

Usage This parameter applies to only z/OS.

TEMPHIE: High water mark of pages used in CCATEMP expansion area

Status Meaning changed in V7R1.0

Class SYSTEM

Default None

Setting View-only

Meaning The TEMPHIE parameter tracks the high water mark of pages used in the CCATEMP expansion area. The expansion area pages start at page number 65,536 (X’10000’).

The TEMPHIE value is reported on the audit trail during EOJ processing in the following message and written in the system final statistics ($$$ SYSTEM = ’7.1.00’) as label TEMX.

```
M204.2622: HWM CCATEMP PAGES USED IN EXPANSION AREA = %C,
MAX AVAILABLE = 19464
```

TEMPHIS: High water mark of pages used in CCATEMP small page area

Status Meaning changed in V7R1.0

Class SYSTEM

Default None

Setting View-only

Meaning The TEMPHIS parameter tracks the high water mark of pages used in the CCATEMP small model page area. The small model page area pages range from page number 0 to 65,535 (X'FFFF').

The TEMPHIS value is reported on the audit trail during EOJ processing in the following message and written in the system final statistic ($$$ SYSTEM='7.1.00') as label TSMX.

```
M204.2622:HWM CCATEMP PAGES USED IN SMALL MODEL AREA = %C,
MAX AVAILABLE = 65536
```

UCPUSLIC: CPU time slice allotment for an individual user

Status New in V7R1.0

Class CWAIT

Default 10

Setting On user's parameter line by system manager

Meaning CPU milliseconds allowed while user is CPU bound.

If a CPU-bound user does not yield the CPU voluntarily after UCPUSLIC milliseconds, the user is time sliced, that is, forced to yield the CPU to the user with the next highest priority.

CPU-bound users that voluntarily yield the CPU are again considered IO-bound.

Increasing this parameter results in improved service for CPU bound requests and reduced service for other requests. A smaller value for this parameter increases the rate at which a CPU-bound request falls in priority, resulting in poorer service for the request.

Although a small value for this parameter improves service for non-CPU-bound requests, it can also reduce overall system throughput and performance by increasing scheduler overhead.

UIOSLIC: I/O time slice allotment for individual user

Status New in V7R1.0

Class CWAIT

Default 30

Setting On user's parameter line by system manager

Meaning The CPU time slice allotment, in milliseconds, for non-CPU-bound users

This parameter determines which requests are considered CPU-bound. A user request is considered IO-bound unless UIOSLIC milliseconds pass without the request voluntarily yielding the CPU. After UIOSLIC milliseconds, the user is considered CPU-bound. From this point, if UCPUSLIC milliseconds expire without yielding the CPU, the user is time sliced (forced to yield the CPU to the user with the next highest priority).

A user voluntarily yields the CPU by any of the following actions:

- Executing a READ SCREEN, READ MENU, or $READ statement
- Requesting I/O to a Model 204 file or external sequential file
- Going into a wait for a record or resource lock

Decreasing IOSLICE causes more requests to be considered CPU-bound, and increasing it causes fewer requests to be classed as CPU-bound.

Increasing this parameter diminishes the impact of dynamic dispatching, thus allowing CPU-bound requests to monopolize the CPU. Decreasing this parameter heightens the effect of dynamic dispatching, resulting in increased scheduler overhead.

For more information about Model 204 dynamic dispatching, see the *Model 204 System Manager's Guide*.

USLCMAX: Number of stop-loop-checks before CSLICE is invoked

Status New in V7R1.0

Class CWAIT

Default 50

Setting On user's parameter line by system manager

Meaning This counter is decremented each time a user reaches the bottom of a FOR loop, REPEAT loop, performs a backward JUMP TO, and on a number of other

occasions. When that number reaches zero, a call is made to the CLSICE (Check SLICE) routine to see if the user has exceeded any maximums—MCPU, MDKRD, MDKWR, MUDD, MCNCT, or MOUT—and should be given the LONG REQUEST message or put into a wait (USLCWAIT). Decreasing this number increases scheduler overhead since CSLICE will be called more often. Increasing the number reduces scheduler overhead, but allows users to run longer before being checked for exceeding maximums.

USLCWAIT: Sleep time at minimum dispatching priority

Status New in V7R1.0

Class CWAIT

Default 0

Setting By system manager

Meaning The elapsed time, in milliseconds, a user will wait after reaching the bottom of his priority range. He will not be rescheduled for execution until this time has elapsed. If no other work is found for Model 204 to process, Model 204 will enter an operating system wait until this user or some other user becomes eligible to be dispatched. In the meantime, other address spaces, virtual machines, or partitions may be scheduled by the operating system.

XAUTOINC: Number of unused pages maintained in Table X by automatic increase

Status New in V7R1.0

Class TABLES

Default 0

Setting During file creation and/or reset by file manager

Meaning The number of unused pages to maintain in Table X. Setting XAUTOINC to a nonzero value indicates that a file's Table X may be automatically increased when the file becomes full due to a lack of space in Table X.

The number of unused pages to maintain for Table X is calculated as the difference between the number of pages in Table X (XSIZE) and the highest Table X page used (XHIGHPG). The minimum number of pages is zero, the maximum is 536,870,911.

XHIGHPG: Table X highest active page

Status New in V7R1.0.

Class TABLES

Default None

Setting View-only

Meaning The highest active Table X page.

XQLEN: Table X queue length

Status New in V7R1.0.

Class TABLES

Default None

Setting View-only

Meaning The current number of pages in the Table X reuse queue. When record deletions occur in a Table X page, the page is added to the reuse queue if the following conditions are true:

- Enough extra bytes of free space are available beyond XRESERVE to satisfy the free space required for the XREUSE parameter.
- At least one record number is available.

For more information about the reuse queue, refer to the *Model 204 File Manager's Guide*.

XRECPPG: Table X extension record slots per page

Status New in V7R1.0.

Class TABLES

Default 256

Setting During file creation

Meaning The maximum number of extension records per Table X page.

The product of XRECPPG and XSIZE must not exceed 536,870,911.

XRESERVE: Table X reserve space per page

Status New in V7R1.0

Class TABLES

Default 17

Setting During file creation or reset by file manager

Meaning The number of bytes on a Table X page which must be free in order to store a new extension record on the page. This parameter should be set to the average extension record length, which can be determined using the TABLEX RECLEN command. The parameter effectively reserves space for the expansion of existing records on a Table X page. The minimum space reserved is 0 bytes; the maximum is 6,140 bytes.

XREUSE: Free space required to reuse Table X page

Status New in V7R1.0.

Class TABLES

Default 20

Setting During file creation or reset by file manager

Meaning The amount (percentage) of free space required on a Table X page before it can be added to the reuse queue. The minimum percentage is zero, the maximum is 100.

Usage To make page free space eligible for the reuse queue sooner, lower the XREUSE value.

XREUSED: Extension slots reused in Table X

Status New in V7R1.0.

Class TABLES

Default 0

Setting View-only

Meaning The number of extension records added to the file that reused the record number of a previously deleted extension record.

XSIZE: Pages in Table X

Status New in V7R1.0.

Class TABLES

Default 0

Setting During file creation

Meaning The number of pages in Table X.

XSIZE can range from 0 to 536,870,911. Use the INCREASE TABLEX command to change the size of Table X.

CHECKPOINT: Requesting and handling a checkpoint

Status Support for CHKPRIV parameter

Privileges Any user can issue a CHECKPOINT command without arguments; a system manager or the operator at the console can issue a CHECKPOINT command with arguments. A system administrator may be able to issue a CHECKPOINT command with certain arguments, dependent upon the setting of the CHKPRIV parameter.

Function Requests that Model 204 perform a checkpoint

Syntax

```
CHECKPOINT [TRAN | SUBTRAN | ABORT | MESSAGE]

 [[SET | UNSET | END] EXTENDED QUIESCE]]
```

Where:

- CHECKPOINT command with no arguments, CHECKPOINT TRAN, and CHECKPOINT SUBTRAN specify by default or explicitly, the type of checkpoint request.

  Any logged in user can issue a CHECKPOINT command with no arguments to attempt a transaction checkpoint.

  A system manager can issue any form of the command.

  The setting CPTYPE is not used by this command.

  Any system administrator can issue one or more forms of the command, if allowed by the setting of CHKPRIV, a non-resettable CCAIN parameter.

- TRAN, the default, specifies to take only a transaction checkpoint.
- SUBTRAN specifies to take a transaction or sub-transaction checkpoint. See the *Model 204 System Manager's Guide* to determine whether you should consider this option and what else is required of your set up.
- ABORT keyword aborts a pending request for a checkpoint.
- MESSAGE keyword displays the status of the most recent checkpoint.
- SET EXTENDED QUIESCE keywords place the Online into an extended quiesce immediately after the next successful checkpoint. Once placed into an extended quiesce, no file updating can take place until the extended quiesce ends. The checkpoint that is taken at the end of Model 204 initialization or recovery is not available for extended quiesce processing.

  The command is ignored if the Online is already in an extended quiesce. Issuing the command multiple times is the same as issuing it once.

  If you define a ring stream journal that has an OFFLOAD dataset defined, then when the extended quiesce begins, Model 204 issues internally an OFFLOAD STREAM command for the ring stream journal.

- UNSET EXTENDED QUIESCE keywords reverse the effect of the SET EXTENDED QUIESCE option. The command is ineffective if the:
  - Online is in an extended quiesce state
  - CHECKPOINT SET EXTENDED QUIESCE command has not been issued

  Issuing this command multiple times is the same as issuing it once.

  Issuing a CHECKPOINT UNSET EXTENDED QUIESCE command when the Online is not yet available for extended quiesce processing, results in the following error:

  ```
  M204.2612: CHECKPOINT COMMAND UNSUCCESSFUL - reasons
  ```

- END EXTENDED QUIESCE keywords terminate an extended quiesce. This command also restarts the checkpoint pseudo subtask and file updating can resume in the Online.

Usage notes

The CHECKPOINT command requests that a checkpoint be performed. The Model 204 checkpoint facility provides a means of recovering a valid copy of a database in case of a system failure. The checkpoint facility operates by logging images of any changed file pages to a checkpoint dataset.

When a checkpoint is performed, updating is temporarily suspended, the database is brought to a valid state, and marker records are written on the dataset. If a subsequent system crash occurs, the database can be rolled back in time to its status at the time of a previous checkpoint. For a detailed explanation of checkpoints and system recovery, refer to the *Model 204 System Manager's Guide.*

When a CHECKPOINT command is executed, Model 204 does not close any currently open files or groups. Model 204 automatically generates a unique identifier to be associated with the checkpoint.

After the checkpoint has been performed successfully, Model 204 informs the operator and the audit trail of the date and time that the checkpoint was completed.

The security required for the ABORT and MESSAGE keywords is the same as for the CHKABORT and the CHKMSG commands.

When a CHECKPOINT command with an END, SET, or UNSET keyword is issued in a valid context, the following message is displayed:

```
M204.2611: CHECKPOINT SET/UNSET/END COMMAND SUCCESSFUL
```

When a CHECKPOINT command with a SET, UNSET, or END keyword is issued in an invalid context, the following message is displayed:

```
M204.2612: CHECKPOINT SET/UNSET/END COMMAND UNSUCCESSFUL
- reason
```

COMPACTB: Invoking the data compactor

Status The DELETE option was added in V7R1.0

Privileges File manager

Purpose Combine as many extension records as possible into one extension record for the current or specified, unordered file.

Syntax

```
COMPACTB [FROM ssss][TO eeee][FREE nn][MAXE nn][DELETE]
```

Where

FROM ssss indicates the starting logical record number where the compactor starts looking for extensions. The default is zero.

TO eeee indicates the ending logical record number where the compactor stops working.

FREE nn indicates the percentage of unused or free pages in Table B or Table X that may be used by the data compactor for new extension records. The default is 10 percent. The percentage of free pages is calculated as follows:

- Table B: `((BSIZE - BHIGHPG) / BSIZE) * 100`
- Table X: `((XSIZE - XHIGHPG) / XSIZE) * 100`

MAXE nn specifies the percentage of a page size (6144 bytes) that defines the maximum extension record size eligible for compaction. Larger extensions are not moved. The default is 80 percent.

DELETE specifies to physically delete logically deleted records. The default behavior is to not delete the logically deleted records.

Usage The data compactor is a CPU and I/O intensive process. CCA recommends that you do not compact data at your site during high load periods. To prevent monopolizing system resources, the data compactor checks for a bump condition and, also, allows higher priority users to run at every 30 records read, regardless of whether the records are compacted or not. The data compactor commits at every record compacted.

If your system crashes, only one record compaction may be lost. Data is never lost. The worst that may happen when the system crashes is that after recovery processing, one record—and only one—will have either the:

- Old extension chain with the new extension chain being orphaned, or
- New extension chain with the old extension chain being orphaned

You cannot reclaim space for orphaned chains without a file reorganization. There is no back out for the data compactor.

The compactor takes preimages of all the pages it will change and writes a journal record containing all compacted extensions for each compaction, which may require that you increase the checkpoint and journal dataset sizes.

There is no restriction on record length or number of extension records for the data compactor, as long as there is enough free space on pages to hold the compacted records.

The COMPACTB command maintains the same logical order—that is, visible to programs—of extension records in the compacted records as in the original record. Their physical order—that is, the order of the page numbers—may differ.

Setting the DELETE argument

To use the DELETE argument of the COMPACTB command for a file, that file must already have a Table X defined for it—XSIZE is greater than zero in the file CREATE arguments.

Setting the MAXE argument

The larger an extension, the less likely that it will be combined with other extensions, because the largest single extension record may not be larger than a page size. When extension chains are very long and contain mainly very short extensions, a smaller MAXE setting may produce better results more quickly. You might want to test various settings to find what percentage is most advantageous for your data.

Running the data compactor

The data compactor tries to lock records on a one-at-a-time basis. Records with extensions that are subject to compaction are locked exclusively as long as compaction for the record takes place. If record lock is not available, then the record is skipped. Avoid running the data compactor with applications that lock large numbers of records for a long time.

Reviewing data compactor statistics

The data compactor combines extension records and stores them as new extensions. The following example has seven records, each with two extensions. For each record, one new extension was created from the two original ones and the two original extensions were deleted. Hence, there were seven additions and 14 deletions.

Example: file statistic changes

```
>IN TEST 1 VIEW BHIGHPG,EXTNADD,EXTNDEL,BQLEN

BHIGHPG   39          TABLE B HIGHEST ACTIVE PAGE
EXTNADD   42          EXTENSION RECORDS ADDED
EXTNDEL   15          EXTENSION RECORDS DELETED
BQLEN     2           TABLE B QUEUE LENGTH
```

```
   IN TEST1 COMPACTB FROM 0 TO 9999999 FREE 100 MAXE 100
NUMBER OF BASIC RECORDS PROCESSED:             20
NUMBER OF EXTENSION RECORDS BEFORE COMPACTION: 27
NUMBER OF EXTENSION RECORDS AFTER COMPACTION:  20
NUMBER OF NOT PROCESSED (LOCKED) RECORDS:      0
NUMBER OF FREE PAGES USED:                     2
   IN TEST1 VIEW BHIGHPG,EXTNADD,EXTNDEL,BQLEN
BHIGHPG   41          TABLE B HIGHEST ACTIVE PAGE
EXTNADD   49          EXTENSION RECORDS ADDED
EXTNDEL   29          EXTENSION RECORDS DELETED
BQLEN     5           TABLE B QUEUE LENGTH
```

At the end of processing, the compactor prints the following statistics:

```
M204.2749: NUMBER OF BASIC RECORDS PROCESSED:                nnnn
M204.2750: NUMBER OF EXTENSION RECORDS BEFORE COMPACTION:    nnnn
M204.2751: NUMBER OF EXTENSION RECORDS AFTER COMPACTION:     nnnn
M204.2752: NUMBER OF NOT PROCESSED (LOCKED) RECORDS:         nnnn
M204.2753: NUMBER OF FREE PAGES USED:                        nnnn
M204.2754: NUMBER OF DELETED LOCIGALLY DELETED RECORDS:      nnnn
```

The value of BQLEN parameter also changed. When the original extensions were deleted, a page became suitable for reuse and was placed on the Reuse Queue. In this example, unused pages were used for new extensions. If new extensions are placed on a page from the Reuse Queue, BQLEN may decrease.

Example: compacting extensions

A single record has nine extension records with the following length. The lengths of the extensions in this example were chosen arbitrarily to illustrate how and when extensions are combined, or not combined.

Ext.1	Ext.2	Ext.3	Ext.4	Ext.5	Ext.6	Ext.7	Ext.8	Ext.9
40	1200	2400	3200	4300	2300	60	90	120

The chain is reduced to four parts:

Part 1: Extensions 1,2, and 3 have a combined length of 3640 that compacts into one extension record.

Part 2: Extension 4 is not moved, because combined with previous extensions it will not fit the page.

Part 3: Extension 5 is not moved, because it cannot be combined with either Extension 4 or Extension 6.

Part 4: Extensions 6,7,8, and 9 have combined length of 2570 that compacts into one extension record.

After compacting, there are four extensions instead of nine.

COMPACTE: Table E compactor to reduce fragmentation

Status A new command in V7R1.0 to compact Table E page gaps between large objects

Privileges File manager

Purpose Combine as many page gaps between large objects as possible into contiguous, free Table E pages.

Syntax

```
COMPACTE FROM nn TO nn EXCL MAXPR nn
```

Where:

- FROM *nn* specifies the beginning segment number. The default is 0.
- TO *nn* specifies the ending segment number. The default is EHIGHPG segment.
- *EXCL* specifies an enqueue on file in exclusive mode and does not do constraints checking to speed up compaction process. The default is no.
- *MAXPR* specifies a number of page reads before the command allows a checkpoint and whether the user is being bumped or exceeding limits, such as I/O, CPU slices, or a higher priority user.

Usage To effectively compact Table E, CCA recommends running a TABLEE command with the SEG option, identifying segments with large number of gaps, running COMPACTE command for segments of interest, and then running another TABLEE command for compacted segments to check the results.

Example The following is an example of COMPACTE output.

```
COMPACTE MAXPR 99 FROM 0 TO 10 EXCL
  ***  M204.2811: NUMBER OF MOVED OBJECTS: 18327
  ***  M204.2812: NUMBER OF MOVED PAGES:   85849
  ***  M204.2813: NUMBER OF RECORD LOCKING CONFLICTS: 0
  ***  M204.2814: NUMBER OF MULTISEGMENT OBJECTS: 0
```

DECREASE: Decreasing the file size

Status Table X support added in V7R1.0

Privileges File manager

Function Decreases the size of a Model 204 file by reducing a portion of the file space

Syntax `DECREASE {TABLEB | TABLED | TABLEE | TABLEX} n`

Where *n* is the number of pages in the specified file table to be reassigned to free space. The legal value for *n* depends upon the settings of the file parameters. For example, if TABLEX is specified, you can only free pages that are not currently in use; thus *n* must be in the range 1 through (XSIZE - XHIGHPG - 1).

DEFINE DATASET: Large dataset characteristics

Status The LARGE argument is new in V7R1.0.

Privileges System administrator

Function Specifies characteristics of a large Model 204 dataset for a database as a basic sequential file

Syntax

```
DEFINE DATASET name [LIKE other-name] WITH SCOPE=SYSTEM
  [[LARGE] [SAM] [SEQUENTIAL] other options]
```

Where

- *LARGE* indicates a dataset for a database, a database file dump, or a journal that may have more than 64K tracks. (Applicable under z/OS 1.7 or later)

  SEQUENTIAL, which indicates the file organization, and *SAM*, which indicates the file access method, are required with the LARGE option.

- *other options,* as are appropriate, are documented in the *Model 204 Parameter and Command Reference* under the DEFINE DATASET command.

Usage notes The DEFINE DATASET command creates a dataset definition. You can define a large dataset under only z/OS 1.7 or later. The LARGE argument applies to:

- CCAJRNL, CCAJLOG, including GDG and stream-based journals and jlogs
- Large datasets for dumping and/or restoring Model 204 files
- Model 204 file datasets

DELETE: Field

Status Support for DTSFN field introduced in V7R1.0

Privileges File manager

Function Deletes a field from a Model 204 file

Syntax `DELETE FIELD fieldname`

Where *fieldname* is the name of a field in the currently open Model 204 file.

Example

```
DELETE FIELD EMPID
```

Usage notes When a field is deleted, Model 204 removes the definition of the field and all occurrences of data for that field.

Depending on the size of the file and the field type, a DELETE FIELD command might take a long time to complete.

You cannot delete record security fields, sort, or hash key fields. If the specified field is in a Model 204 file with record security, the file manager must have record security override privileges.

When it processes DELETE FIELD, Model 204 ends any update unit in progress and begins a nonbackoutable update unit. For more information about Model 204 update units, see the *Model 204 File Manager's Guide*.

Handling the date/time stamp field in a file

The DELETE FIELD command is prohibited for the DTSFN field in a file when the FOPT=X'10' is set. Attempting to do so results in the following message:

```
M204.2727: CAN'T DELETE DTS FIELD WHEN FOPT=X'10' IS ON
```

FILELOAD: Starting a File Load program (single-step)

Status Support for Table X added in V7R1.0

Privileges File manager (as User 0 only)

Function Starts a File Load (single-step) program

Syntax

```
FILELOAD k, n, m [,[s1] [,[s2] [,[s3] [,[s4]
    [,v1]]]]]
```

Where

- *k* specifies the maximum number of records to be loaded into the Model 204 file.
- *n* specifies the maximum number of times that the statements in the File Load program are to be executed.
- *m* specifies the number of input records to skip before starting the load process.
- *s1* is the amount of memory that the sort uses for sorting the deferred update index records produced by the File Load program. If *s1* is omitted or is not a positive integer, a default value of 50000 bytes (64K for z/VM) is used for the sort work space. It is important to increase *s1* for large sorts. The *s1* argument should be made as large as practical, especially for sorts involving many records.
- *s2* represents the amount of memory to be used for sorting the deferred update FRV index records. It also defaults to 50000 bytes (64K for z/VM) if *s2* is omitted or is not a positive integer. For 3380 disk drives, the minimum values for s1 and *s2* are 64000 bytes.
- *s3* represents the amount of memory allocated for the two buffers that pass data from Model 204 to the sorts. This argument specifies the combined size of the two buffers. The core specified in *s3* is evenly divided between the two buffers, and *s3* defaults to 8000 bytes. The minimum value accepted is 48 bytes, but such a small size is not recommended. Larger sizes might produce a noticeable performance improvement. An increase in *s1* normally has a greater effect on performance than a corresponding increase in *s3*.
- *s4* represents the amount of memory that is allocated for the two buffers used to pass data from the sorts to Model 204. *s4* is ordinarily equal in size to *s3* and is set equal to *s3* by default.
- *vl* specifies the maximum sort record length in variable-length record sorts.

  The *vl* argument is required when ORDERED fields are updated by the File Load program. For efficiency, the *vl* option should not be present when

ORDERED fields are not updated. For more information on the *vl* option, refer to the *Model 204 File Manager's Guide*.

Syntax notes

FILELOAD must be issued in file context (that is, the current default must be a file, not a group).

Specify commas to hold places for missing arguments. When an argument is missing, the default value (if any) is used. (See the following examples.)

Example

```
OPEN CENSUS

FILELOAD -1,-1,1,102400,100
G
 HOUSEHOLD ID=1,5,X'804C'
 SEX=6,1,X'0410'
 UNKNOWN=MALE=FEMALE=


 PLACE SIZE=9,2,X'0410'
 UNKNOWN=
 FARM=RURAL=URBAN=5000=10000=
 25000=50000=100000=250000=
 500000=1000000= OVER 1000000=

END
```

In the following example, arguments s1 and s4 are missing (default values are used):

```
FILELOAD -1,-1,1,,1000,1000
```

Usage notes

The FILELOAD command invokes the File Load (FLOD) utility. It is used when a single-step File Load procedure is being performed.

The FLOD command is used to invoke a multistep File Load procedure.

FILELOAD signals the start of a File Load program, which consists of special statements that load data from a sequential file into a Model 204 file.

To make the File Load program execute efficiently, set the MAXBUF parameter to a value between 5 and 10. Also, make the value of the s1 attribute as large as is practical. The exceptions to these guidelines are:

- A FILELOAD of a hash-order file that does not use the sort exit M204HASH
- A FILELOAD that extensively uses the locate statement

The Model 204 FILELOAD exit feature lets you modify the record image upon which the FILELOAD (or FLOD) command acts. With the FILELOAD exit feature, you can create a COBOL or Assembler program to modify the image records. The FILELOAD exit feature enables the use of more than one input file for FILELOAD processing. The FILELOAD exit is modeled on SORT exits,

similar to the E15 sort exit. The FILELOAD exit feature is documented in the *Model 204 File Manager's Guide*.

When it processes FILELOAD, Model 204 ends any update unit in progress and begins a nonbackoutable update unit. For more information about Model 204 update units, see the *Model 204 File Manager's Guide*.

The FILELOAD command, the File Load utility, the File Load statements, and input dataset requirements are discussed in detail in the *Model 204 File Manager's Guide.*

FLOD: Starting a File Load program (multistep)

Status Table X support added in V7R1.0

Privileges File manager (as User 0 only)

Function Starts a File Load (multistep) program

Syntax `FLOD k,n,m`

File Load phases

The first phase of the multistep File Load is the execution of the File Load program, which formats and loads the data into Table B (and Table X) of the Model 204 file. The first phase also generates the deferred updates for the Model 204 file index (Tables C and D) and places these deferred updates in the deferred update dataset or datasets. The second phase of the multistep File Load consists of sorting each of the deferred update datasets generated in the first phase and the FRV deferred update dataset, if one is generated, and then applying the sorted deferred index updates to Tables C and D.

INCREASE: Increasing the file size

Status Table X support added in V7R1.0, as well as the ability to dynamically issue the command without stopping the file and bumping users.

Privileges File manager

Function Increases the space available for a Model 204 file

Syntax `INCREASE {DATASETS [WITH] ddname [,ddname]...}`

Where

- *ddname* is the dataset whose pages are to be allocated to free space.

  `INCREASE {TABLEB n [DYNAMIC]}`

  `| {[TABLED | TABLEE | TABLEX] n}`

- *n* is the number of pages in free space to be assigned to the specified table. The value of *n* is in the range 1 through the value of the FREESIZE parameter (the number of unassigned pages available in the Model 204 file).
- DYNAMIC is for Table B dynamic increase only. It allows Table B to be increased even if the file is open and has requests compiled against it.

Syntax notes An INCREASE command must be issued in file context. That is, the current default must be a file, not a group.

The argument *ddname* can be separated by commas or by one or more blanks. A *ddname*, not a dataset name, must be used in the command.

Examples

```
OPEN CARS
update password

INCREASE DATASETS WITH CARS1, CAR2, C3, CABC
INCREASE TABLEB 500
INCREASE TABLED 800
```

Usage notes If it is required to expand FREESIZE by adding extra datasets to the file to support the increase of a table, then the INCREASE DATASETS command should precede the INCREASE command for Table B, Table D, Table E, or Table X.

You can assign additional pages from free space for use by either Table B (except for hash key files), Table D, Table E, or Table X. The size of Tables A and C cannot be changed without reorganizing the file.

You can issue INCREASE DATASETS, INCREASE TABLEB, INCREASE TABLED, INCREASE TABLEE, and INCREASE TABLEX commands without

stopping a file and bumping all users out of the file. The file may be accessed or updated at the same time while the INCREASE command is processing.

However, Model 204 handles INCREASE processing only one type at a time in succession, not simultaneously. You cannot do an INCREASE DATASETS and an INCREASE TABLEB at the same time.

You can increase Table B, even when the file is open and has requests compiled against it by using a new option, DYNAMIC. Without this option the INCREASE TABLE B command processes only files that are not open by other users or subsystems. There is a small performance penalty for MP/204 users: multiprocessing is stopped for a short time while internal changes are made.

Increasing and decreasing a file, the various tables that make up a file, and the relevant table arguments are explained in detail in the *Model 204 File Manager's Guide*.

If you are increasing the size of a file in response to a file full condition (FISTAT=X'08'), reset FISTAT to X'00' after performing the INCREASE command. However, make sure that the FISTAT X'02' bit (File is physically inconsistent) is not set. *Never* reset FISTAT for a physically inconsistent file.

Privileges required

To use the INCREASE command you must have update authority to the selected file. If the file is open for read-only access, the INCREASE command is rejected with the following message:

```
M204.1036: UPDATES NOT PERMITTED
```

The INCREASE command may not be used to increase XSIZE from zero.

MONITOR: An Online system

Status Display reformatted in V7R1.0

Status The DISKBUFFG option is introduced in V7R1.0, as well as the display of Table X pages currently in use.

Privileges System administrator

Function Monitors the disk buffers, displaying the number of pages in each table of each file that currently resides in the disk buffer.

Syntax `MONITOR {DISKBUFF | DISKBUFFG | DISKBUFFL}`

Where

- DISKBUFF specifies to return the statistics for both the above and below the bar buffers.
- DISKBUFFG specifies to return the statistics for the above the bar buffer pool usage.
- DISKBUFFL specifies to return the statistics for below the bar buffers.

Example

```
MONITOR DISKBUFF
```

or

```
MONITOR DISKBUFFG
```

Usage notes The synonym M applies to all MONITOR commands. For example, you can enter M DISKBUFF instead of MONITOR DISKBUFF.

The MONITOR DISKBUFF command displays the following types of information, in the format presented below, where FCT is the number of pages in the File Control Table:

MONITOR DISKBUFF

```
FILENAME   FCT     TBLA     TBLB     TBLC     TBLD     TBLE     TBLX    *TOTAL*
--------   ---   ------   ------   ------   ------   ------   ------    -------
CCATEMP      0       40        0        0        0        0        0         40
PROC1        1        0        0        0       11        0        0         12
FILETBLX     1        1       10        0        2        0        2         16
TESTZ        1        1       11        0        3      185        0        201
*TOTAL*      3       42       21        0       16      185        2        269
```

The MONITOR DISKBUFFG command displays above the bar.

MONITOR DISKBUFFG

FILENAME	FCT	TBLA	TBLB	TBLC	TBLD	TBLE	TBLX	*TOTAL*
CCATEMP	0	3	0	0	0	0	0	3
CCAGRP	1	0	0	0	1	0	0	2
CCASYS	1	1	7	8	15	0	0	32
EWDPROC	1	0	0	0	3	0	0	4
EWDSCR	1	0	0	0	1	0	0	2
EWDDATA	1	0	0	0	1	0	0	2
FILETBLX	1	0	0	0	2	0	0	3
TOTAL	6	4	7	8	23	0	0	48

The MONITOR DISKBUFFL command displays below the bar buffer pool usage.

MONITOR DISKBUFFL

FILENAME	FCT	TBLA	TBLB	TBLC	TBLD	TBLE	TBLX	*TOTAL*
CCATEMP	1	3	0	0	0	0	0	3
CCAGRP	1	0	0	0	1	0	0	2
CCASYS	1	1	0	0	11	0	0	13
EWDPROC	1	0	0	0	3	0	0	4
EWDSCR	1	0	0	0	1	0	0	2
EWDDATA	1	0	0	0	1	0	0	2
FILETBLX	1	0	0	0	2	0	0	3
TOTAL	6	4	0	0	19	0	0	29

MONITOR DATASPACE: Dataspaces and hiperspaces

Status

The MONITOR DATSPACE command has been enhanced to include a new type of allocation: 64-BIT MEMORY. These data areas will be allocated above the bar under a z/OS or z/VM system and not in dataspaces. These allocations will occur when all of the following are true:

1. No CCASERVR DDs are allocated
2. TEMPPAGE is non-zero
3. APSYPAGE is non-zero

Privileges

System administrator

Function

Enables the system administrator to display the use of dataspaces and hiperspaces when tracking the use of the CCATEMP and CCASERV In Storage feature.

Syntax

```
MONITOR DATASPACE
```

Example

The DATASPACE option of the MONITOR command displays dataspace and hiperspace information as shown in the following example. ***Hiperspace*** is an area of virtual storage requested from the hardware's expanded storage. Access to hiperspace saves you from paging out to disk.

```
NAME      STORAGE TYPE         4K PAGES   PAGE HWM EXTRA DATASPACES
--------  ------------------- ---------- ---------- ----------------
CCASERVR 64-BIT MEMORY             1465       1440
CCATEMP  64-BIT MEMORY            15100          0
CCAAPSY  64-BIT MEMORY             1000          0

NAME      READS  WRITES  PAGES READ  PAGES WRITN  SLOWRD  SLOWWR  PAGEF
--------  ------ ------- ----------- ------------ ------- ------- ------
CCASERVR     13      20        1261         1940       0       0      0
CCATEMP       0       0           0            0       0       0      0
CCAAPSY       0       0           0            0       0       0      0
```

MONITOR SIZE: Tracking journal and checkpoint streams

Status New in V7R1.0

Privileges System administrator

Function The MONITOR SIZE command returns a screen display of the current definitions for a journal or checkpoint stream along with the current usage.

Syntax

```
MONITOR SIZE {[JOURNAL | JRNL] | [CHECKPOINT | CHKP] |
CCAJRNL | CCAJLOG | CHKPOINT | CHKPNTS}
```

Where

- *JOURNAL* (or *JRNL)* specifies to display the definitions and current usage for the CCAJRNL and CCAJLOG streams.
- *CHECKPOINT* (or *CHKP)* specifies to display the definitions and current usage for the CHKPOINT and CHKPNTS streams.

Example

```
MONITOR SIZE CHKP
```

```
 DDNAME        TYPE                 TRKS_ALLOC    TRKS_USED     PCT_USED
_________ ____________________      __________    _________     ________
CHKPOINT (GDG STREAM)
CHK1     (DATASET)                         100           40          40%
              CCA008\M204V710.CHKPOINT.G3371V00\+5

CHKPNTS  (DATASET)                        5000         3000          60%
              CCA013\M204V710.CHKP5
```

In the previous example, CHKPOINT is defined as a GDG STREAM consisting of member CHK1, a GDG dataset. The volser, GDG dataset name and relative generation are displayed.

CHKPNTS is defined as a simple dataset. Its volser and dataset name are also displayed.

NOTE: If CPMAX=1 and a checkpoint occurs, the CHKPOINT dataset, in this case CHK1, is repositioned to the beginning of the dataset (the first dataset if a GDG) and the TRKS_USED is reset to zero.

When a sub transaction checkpoint is taken, a switch from CHKPOINT to CHKPNTS or vice-versa occurs and the target of the switch is repositioned to the beginning of the dataset and TRKS_USED is reset to zero but starts increasing immediately. The switched from member has TRKS_USED set to zero.

In addition, you can restrict the output of the MONITOR SIZE command to only the one specified stream, for example:

```
MONITOR SIZE CCAJLOG
```

MSGCTL: Setting message output

Status In V7R1.0 the hierarchical structure of MSGCTL is reinstated and message types—ER, RK, AD, and MS—are mutually exclusive.

Privileges System administrator

Function Specifies the actions to take when Model 204 produces a particular error or informational message

Syntax

```
MSGCTL {M204 | USER}.msg-number
 [msg-option [msg-option]... ]
 [CLASS={P | I | E}] [RETCODEO=online-retcode]
 [RETCODEB=batch-retcode]
```

Where

- *M204* specifies a standard Model 204 message.
- *USER* specifies a user message.
- *msg-number* is the four-digit number of the message.
- *msg-option* is one of the following:

Table 6-1. MSGCTL command message options

Option	Action…
AUDITAD	Puts specified message on the audit trail as an AD line.
AUDITER	Puts specified message on the audit trail as an ER line.
AUDITMS	Puts specified error message on the audit trail as an MS line.
AUDITRK	Puts specified message on the audit as an RK line.
COUNT	Increments, by one, the message count whenever this message is issued. If the message count exceeds the value of the ERMX parameter, the user is restarted.
DUMPALL	Dumps entire Model 204 region. Provides the maximum amount of information for problem determination.
NOACTION	Ignores original option that was assigned the message and returns to main processing.
NOAUDIT	Suppresses auditing of a specific error message.
NOAUDITAD	Changes the specified AD message on the audit trail to an MS line.
NOAUDITER	Changes the specified ER message on the audit trail to an RK line.

Table 6-1. MSGCTL command message options (continued)

Option	Action…
NOAUDITMS	Suppresses auditing the specified MS message.
NOAUDITRK	Changes the specified RK message on the audit trail to an AD line.
NOCOUNT	Does not increment, by one, the message count whenever this message is displayed.
NODUMP	Does not generate a dump.
NOOPR	Does not display specified message on the operator's console.
NOPREFIX	Suppresses display of the message prefix and number.
NOSAVE	Does not save specified message in the message save table. The user should note that messages sent to the operator and messages that cause SNAPs are always added to the message save table, regardless of the selection of the SAVE or NOSAVE options.
NOSNAP	Suppresses the production of a SNAP when the specified error condition occurs.
NOTERM	Does not display specified message on the user's terminal or for single user jobs in CCAPRINT. **Note:** Model 204 passes return codes back in login and logout messages; if you set the NOTERM option for login and logout messages you cannot get the return code.
OPR	Displays specified message on the operator's console.
PREFIX	Displays a message prefix (M204 or USER) and message number along with the message.
SAVE	Puts a copy of the specified message on the message save table whenever that message is issued. Use the VIEW ERRORS command to examine the saved message.
SNAP	SNAP produces the same information as SNAPALL.
SNAPALL	SNAP includes all allocated storage except the Model 204 load module.
SNAPPDL	SNAP includes the registers, module map (alpha and address order), allocated storage map, pushdown list trace, user's server, KOMM, disk buffers containing Model 204 file pages held by the current user (maximum of 4), file directory of current file, and patch information.
SNAPSEL *n*	SNAP includes everything in SNAPPDL as well as other items specified by *n.* The settings for n can be added together. For example, to output the disk buffers for Tables B and C: `MSGCTL SNAPSEL 6`

Table 6-1. MSGCTL command message options (continued)

Option	Action…
where *n*=1	Output all Table A and FCT disk buffers for the current file and no CCATEMP disk buffers.
where *n*=2	Output all Table B disk buffers for the current file and no CCATEMP disk buffers.
where *n*=4	Output all Table C disk buffers for the current file and no CCATEMP disk buffers.
where *n*=8	Output all Table D disk buffers for the current file and no CCATEMP disk buffers.
where *n*=16	Output the disk buffers requested by the previous settings for all files open by this user and no CCATEMP disk buffers. This setting directs the buffer output and coordinates with one of the previous settings.
where *n*=32	Output all data structures except those which pertain to record locking, resource locking and disk buffers.
where *n*=64	Output those data structures which pertain to record locking and resource locking.
where *n*=128	Output all CCATEMP disk buffers and no Model 204 file disk buffers.
TERM	Causes the specified message to be displayed on the user's terminal or for single user jobs in CCAPRINT.

- *CLASS* arguments are used on only messages that have been suppressed in an application subsystem (APSY). The CLASS arguments have no effect on messages appearing outside of an APSY. For example, if an APSY suppresses *E* (error) messages, you can reset the message to *I* (informational). It will then appear at the user terminal. If both *I* and *E* are suppressed, you can reset it to *P* (prompt) and the message will appear at the terminal.
  - *CLASS=P* resets the message so that it is treated as a prompting message. Since you cannot suppress the printing of prompting messages at the terminal, the message is displayed at the APSY user's terminal.

    Note: *P* does not cause a prompt; it only forces the message to appear at the terminal in an APSY.
  - *CLASS=I* (informational) prevents the specified message from being printed at the terminal within an APSY for which informational messages are suppressed.
  - *CLASS=E* (error) prevents the specified message from being printed at the terminal within an APSY for which error messages are suppressed.

- *RETCODEO* specifies a job step return code in the range 0 to 255 to be associated with the message in Online runs. The actual job step return code is the maximum return code of all messages issued.
- *RETCODEB* specifies a job step return code in the range 0 to 255 to be associated with the message in batch runs. The actual job step return code is the maximum return code of all messages issued.

See the *Model 204 System Manager's Guide* for a description and listing of job step return codes. When there have been multiple errors, the highest return code is displayed. The *Model 204 Messages Manual* documents the Online and Batch return codes for each message.

Syntax notes

A period with no space before or after must be included between the message prefix, M204 or USER, and the message number.

Any number of nonconflicting options can be specified in the command. If conflicting options are specified, for example, PREFIX and NOPREFIX, the MSGCTL option specified last takes effect.

You can specify the MSGCTL command before the User 0 parameter line in the CCAIN stream. This lets you format messages displayed during initialization and recovery.

The MSGCTL message options—AUDIT, AUDITAD, AUDITMS, AUDITRK—are mutually exclusive. If you enter multiple options for a message, the last option listed is processed.

The COUNT option is a separate process and can be used with the AUDIT, AUDITAD, AUDITMS, or AUDITRK option, as needed.

Hierarchical evaluation of message types

The hierarchical structure of messages types is as follows:

- The NOAUDITER option lowers an ER message to an RK message type.
- The NOAUDITRK option lowers an RK message to an AD message type.
- The NOAUDITAD option lowers an AD message to an MS message type.
- The NOAUDITMS option lowers an MS message, so that it is not audited, as though you had used the NOAUDIT option.

The NOAUDITxx option only processes a message of the 'xx' type. However, NOAUDIT works on any message type.

If NOAUDITxx is specified for a message that is not of type 'xx', the following message is returned:

```
M204.1047 NOAUDITxx IS AN INVALID PARAMETER FOR THIS
MESSAGE
```

Example

The following example suppresses the display of the message prefix and number associated with the standard Model 204 password prompt:

```
MSGCTL M204.0347 NOPREFIX
```

Usage notes

The MSGCTL command lets the system manager change the way Model 204 processes a particular error or informational message. Changes made to a message affect all users in the Online.

You can use the MSGCTL command to change processing for either standard Model 204 messages with a Model 204 prefix or for messages with a user prefix developed by the installation. Note that you can also use the MSGCTL parameter to control the display of message prefixes as well as error and informational messages on the user's terminal. User messages are built in the MSGU module distributed with Model 204 and described in the *Model 204 Installation Guide*.

Model 204 handles different message event conditions in different ways. For example:

- A message can be displayed on the operator's console.
- A message can be displayed on the user's terminal.
- A message can be written to the journal or audit trail dataset.
- An error can result in the production of a SNAP.

For all Model 204 message event conditions, the *Model 204 Messages Manual* lists the error number, the text of the message, and the default action that is taken when the event occurs.

Understanding the NOCOUNT option

Caution: CCA strongly recommends that you do not change or suppress the COUNT option that is assigned to some messages. The User Language compiler depends on using the COUNT option to track errors in your program. If no errors can be reported, your program will seem to compile only to abend during the run. Also, the lack of messages in your Model 204 journal, audit trail, or other job output can result in the inability to trace and diagnose problems.

How the MSGCTL command can override both the MSGCTL and DEBUGUL parameters

If you use the MSGCTL command, and suppress the display of a message by using the NOTERM keyword, then no matter how the MSGCTL or DEBUGUL parameters are set, the message does not go to the terminal. Conversely, if you want messages displayed at the terminal, enter the MSGCTL command using the following syntax:

```
MSGCTL {M204 | USER}.msg-number TERM
```

Understanding the NOACTION option

Error messages have a one-byte indicator that informs message processing whether the message needs to be processed and whether the message has been changed using MSGCTL. If a message does not need processing, Model 204 resumes the main processing.

Note: NOACTION has no effect on a restart or a termination message.

PRIORITY: Assigning a priority class

Privileges System manager, system administrator, or operator

Function Assigns a user to a priority class or displays information about priority assignments

Syntax

```
PRIORITY [usernumber [,LOW | ,STANDARD | , HIGH]]
```

or

```
PRIORITY usernumber [,cur | (cur,min,max)] [,key-
word=value]
```

Where

- *usernumber* is the number of the user whose priority is being assigned or displayed.

If specifying…	Then Model 204 lists priority information…
Neither *usernumber* nor the priority	For all active users (see the format below).
Only *usernumber*	For only that user.
Both *usernumber* and the priority	Specified to that user.

- *user* specifies user number to modify or display. If no further arguments are specified, only the user will be displayed.
- *cur* specifies new current priority for specified user (0-255)
- *min* specifies the new minimum priority for specified user (1-253)
- *max* specifies the new maximum priority for specified user (1-253)
- *keyword* specifies the value to be assigned to the specified user. (To reset values to system defaults, specify a null value, for example IOSLICE=. Zero is a valid value, for example IOSLICE=0. The following keywords are recognized:

Keyword	Meaning
IOSLICE	CPU milliseconds allowed while user is I/O bound
CPUSLICE	CPU milliseconds allowed while user is CPU bound
SLCWAIT	Sleep time in milliseconds when a user reaches minimum priority level.
SLCMAX	Number of SLCs before CSLICE invoked (max=65545). Declines in this number increase the accuracy of the slice interval. *However, CPU overhead increases.*

Syntax notes The comma is optional.

Example The following example requests priority information about all active users. The SERV column is either a server number OUT for swapped out.

```
PRIORITY
```

```
USER USERID   P CUR,MIN-MAX   SLICE  IOSLICE CPUSLIC MAX SLCWAIT SERV     CPU
   0 NO USERI S 253,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.001
   1 BECKETT  S 061,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.013
   2 LESTER   H 127,080-127   0.000I   0.070   0.100  50   0.00     5    0.170
   3 MATSUZAK S 061,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.012
   4 PENNY    H 104,080-127   0.000I   0.070   0.100  50   0.00     1    0.422
   5 WAKEFIEL H 114,080-127   0.000I   0.070   0.100  50   0.00     3    0.105
   6 VARITEK  S 079,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.063
   7 YOUKILIS S 079,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.069
   8 PEDROIA  S 079,032-079   0.000I   0.070   0.100  50   0.00     6    0.133
   9 LOWELL   S 072,032-079   0.000I   0.070   0.100  50   0.00   OUT    0.032
  11 LUGO     S 079,032-079   0.000I   0.070   0.100  50   0.00     2    0.112
```

The following example requests priority information about user number 112:

```
PRIORITY 112
```

The following example assigns a HIGH priority to user number 2:

```
PRIORITY 2 HIGH
```

REGENERATE: Recovering a file in two passes

Status The IGNORE option new in V7R1.0.

Privileges User 0

Function In the event of a media failure, for example, a disk head crash, a REGENERATE command recovers a Model 204 file by reapplying some or all of the updates that have been made to the file since the time of the last dump.

Syntax

```
REGENERATE FILE filename [FROM dumpname | IGNORE]
 [TO {LAST UPDATE [BEFORE yy.ddd hh:mm:ss.th]
 | LAST CHECKPOINT [BEFORE yy.dd hh:mm:ss.th]
 | UPDATE nn OF yy.ddd hh:mm:ss.th
 | CHECKPOINT yy.ddd hh:mm:ss.th}]
```

For more than one file, specify REGENERATE only once; the remaining syntax is repeated for each file.

Where

- *filename* is the name of a Model 204 file that is one to eight characters long.
- A FROM clause determines whether or not REGENERATE processing performs a restore:
  - If a FROM clause is present for a file, REGENERATE processing restores that file from the Model 204 dump specified in the clause.
  - If a FROM clause is not present for a file, REGENERATE processing assumes that the file was previously restored.
- *dumpname* is the ddname of the dumped file, one to eight characters long, from which to recover.
- *IGNORE* lets REGENERATE processing bypass the file parameter list (FPL) update timestamps, so you can run REGEN processing with one CCAGEN at a time, instead of requiring you to concatenate the CCAGEN files.
- The TO clause options listed in Table 6-2 identify the stopping point:

Table 6-2. TO clause options

Option	Applies updates to the file up to…
LAST UPDATE	All. This is the default.
LAST CHECKPOINT	Last checkpoint on the input journal.

Table 6-2. TO clause options (continued)

Option	Applies updates to the file up to…
UPDATE number [OF *id*]	And including the specified update. The ID, which is the date/time stamp from the END OF UPDATE message, is specified in the form: *yy.ddd hh:mm:ss.th.*
CHECKPOINT *id*	Specified checkpoint. The ID is specified in the form: *yy.ddd hh:mm:ss.th.*

- A BEFORE clause specifies a nonspecific stopping point. The BEFORE clause is an alternative to specifying an exact checkpoint ID or update unit number and ending date and time from an audit trail.

  If you specify a BEFORE clause, Pass One of REGENERATE processing determines the last complete update or checkpoint before the time specified. Pass Two of REGENERATE processing stops reapplying file updates when it reaches the stopping point identified in Pass One.

Syntax notes When multiple files are regenerated, each file must be specified on a separate line. The line can begin in any column. The END keyword indicates the end of the list of files.

Examples

```
REGENERATE FILE DB1 FROM DUMPDB1

REGEN FILE DB1 FROM DUMPDB1 TO LAST UPDATE

REGEN FILE DBAUDIT FROM DUMP01 TO LAST -
CHECKPOINT

REGEN FILE DB1 FROM DUMPDB2 TO UPDATE 3 -
OF 88.216 06:28:42.98

REGEN FILE MODFILE FROM DUMPMOD TO -
CHECKPOINT 88.216 01:05:22.99

REGEN
  FILE DB1 FROM DUMPDB1 TO LAST CHECKPOINT
  FILE DB2 FROM DUMPDB2
  FILE DB3 FROM DUMPDB3 TO UPDATE 5 OF -
  88.008 05:19:41.27
END
```

Note: Blank lines are not valid between the REGENERATE and END command.

```
REGEN FILE ABC FROM DUMPABC /* First run */
.
.
REGEN FILE ABC IGNORE          /* A later run */
```

Usage notes

The REGENERATE command lets the system manager use the Model 204 media recovery feature to regenerate one or more files. A REGENERATE command is issued in case of a loss of storage integrity caused by a hard system error such as a disk head crash.

The algorithm for identifying starting points, used whether or not a FROM clause is present, is to find the first start of an update unit with a time greater than the last updated time of the file being processed.

If REGENERATE processing opens a file for which it is *not* performing a restore, and if that file is marked physically inconsistent, REGENERATE processing for that file is discontinued.

When it processes REGENERATE, Model 204 ends any update unit in progress and begins a no-back out-allowed update unit. For more information about Model 204 update units, see the *Model 204 File Manager's Guide*.

For a single-user (batch) Model 204 run, specify a REGENERATE command on User 0's command line. REGENERATE need not be the first command issued. You can intermix User Language procedures and other batch functions with REGENERATE commands.

Using the IGNORE option

For example, providing the first journal the first REGEN command might be

```
REGEN FILE ABC FROM DUMPABC
```

A second run may provide the second journal and specify:

```
REGEN FILE ABC IGNORE
```

The second run begins where the previous REGEN processing ended and applies the second journal updates.

Caution: If you omit a journal, Model 204 does not report this to you. Therefore, use the IGNORE option with care.

Using REGENERATE with multiple files

The number of files you can regenerate in a single run depends upon the settings of the NFILES and NDIR parameters. See the *Model 204 Parameter and Command Reference.*

RENAME FIELD: Renaming a field

Status Date/time stamp support added in V7R1.0

The RENAME FIELD command is prohibited for the DTSFN field in a file when the FOPT=X'10' is set. Attempting to do so results in the following message:

```
M204.2727: CANT DELETE DTS FIELD WHEN FOPT=X'10' IS ON
```

SWITCH STREAM: Change stream to next member

Status New in V7R1.0

Privileges System manager

Function You can switch streams—parallel, ring, concatenated, GDG—to the next member at any time.

Syntax

```
SWITCH STREAM streamname
```

Where *streamname* specifies CCAJRNL, CCAJLOG, CHKPOINT, or CHKPNTS. The command is valid for only these streams.

Usage notes If you specify:

- Journal stream CCAJRNL or CCAJLOG, the currently active dataset is closed and the next dataset defined to the stream is opened.
- Checkpoint stream CHKPOINT or CHKPNTS, the currently active dataset in the stream is closed after the next record is written to the dataset.

If a SWITCH STREAM CCAJRNL command is issued during CHECKPOINT EXTENDED QUIESCE command processing, the journal will be switched, a new checkpoint issued, and the duration extended quiesce time-out set in CPQZSECS will be reset to its maximum, original value—disregarding the already elapsed time.

TABLEB: Determining usage of Table B

Status Support for Table X added in V7R1.0

Privileges File manager

Function Displays information about page use in Table B of a Model 204 file

Syntax

```
TABLEB [LIST] [RECORD LENGTH] [OVERFLOW] [MASTER]
 [PAGE pagenumber]
 [PAGES pagenumber1 TO pagenumber2]
```

Where The following options display space utilization use in Table B. A TABLEB command accepts only one option per command, except if the LIST option is specified, which may be followed by the RECORD LENGTH option.

Option	Displays…
LIST	One line for each Table B page scanned, from the lowest to the highest page.
RECORD LENGTH	Average length of a logical record in Table B. Or, average length of base record in Table B if Table X is defined for the file. Issue a TABLEBX command, if you want the average length of a logical record, when Table X is defined. RECORD LENGTH can be abbreviated as RECLEN.
OVERFLOW	Table B averages for the overflow area of a sorted file.
MASTER	Table B averages for the master area of a sorted file.
pagenumber, pagenumber1, and *pagenumber2*	Specific pages in Table B.
PAGE	Table B averages for the specified file page.
PAGES	Table B averages for page *number1* through page *number2*.

Example

```
TABLEB
507 AVG. FREE SPACE PER PAGE
  2 AVG. FREE SLOTS PER PAGE
 68 NUMBER OF PAGES PROCESSED
 16 BRECPPG - TABLE B RECORDS PER PAGE
 17  BRESERVE - TABLE B RESERVED SPACE PER PAGE
```

Usage notes You can issue a TABLEB command only in file context— that is, the current default must be a file, not a group.

TABLEB command output can be routed to a USE dataset.

If you specify a TABLEB RECLEN *x* command, where *x* is something other than end-of-line, the following error is issued:

```
M204.0643: AN OPERAND IS INVALID OR MISSING
```

Using the TABLEB command without any option provides only the averages (shown previously) over Table B active pages. For a sorted file, the averages are computed separately for the overflow and master pages, and they appear separately in the output.

Using the LIST option

To get the exact amounts of free space and free slots on each active page, you must use the LIST option. This option produces a printed line for each Table B page scanned, starting with the lowest active page and ending with the highest one. The output of the LIST option, which precedes the averages shown above, is displayed in the following format:

```
PAGE NO.  FREE SPACE  FREE SLOTS

nnn       nnn         nnn
.         .           .
.         .           .
.         .           .
```

If the current file is a sorted one, each line includes an indication of the type of page that the line describes. For example, if page number 5 is an overflow page, the output might be:

```
PAGE NO.  FREE SPACE  FREE SLOTS
5         200         28        OVERFLOW
```

Limiting the range of pages scanned

You can limit the range of Table B pages to be scanned in the following ways:

- Specify the PAGE or PAGES option to select a single page or a range of pages.

  The following command causes a scan of five pages, if they fall in the range of active pages. If any of the selected pages is inactive, an error message is issued and the command is ignored. Page numbers are specified in decimal and cannot be negative. For example:

  ```
  TABLEB PAGES 0 TO 4
  ```

- Limit the range of pages scanned. This way applies only to sorted files. Use the options OVERFLOW or MASTER. In either case, the scan is done only over the specified pages.

Using the RECORD LENGTH option

The RECORD LENGTH option provides an estimate of the average length of a logical record in Table B, if Table X is not defined for the file (XSIZE=0). A logical record is a record that includes all its extensions.

If Table X is defined for the file (XSIZE>0), a TABLEB command provides an estimate of the average length of the base record in Table B.

The average record length is exact unless DELETE RECORDS or IFDSET is used in the file. If DELETE RECORDS or IFDSET is used, the record length computed is too big, because space from the deleted records is never released.

The RECORD LENGTH option is available only when all of Table B is scanned. Or, to state it another way, RECLEN option does not work with either the PAGE *n* or the PAGES *n1* TO *n2* option.

TABLEBX: Determining use of Tables B and X

Status New in V7R1.0

Privileges File manager

Function Displays information about page use in Tables B and X of a Model 204 file

Syntax

```
TABLEBX [LIST] [RECORD LENGTH] [OVERFLOW] [MASTER]
 [PAGE pagenumber]
 [PAGES pagenumber1 TO pagenumber2]
```

Where The following options display PAGE use in Tables B and X. A TABLEBX command accepts only one option per command, except if the LIST option is specified, which may be followed by the RECORD LENGTH option.

Option	Displays…
LIST	One line for each Table B and Table X page scanned, from the lowest to the highest page.
RECORD LENGTH	Average length of a logical record in Tables B. RECORD LENGTH can be abbreviated as RECLEN.
OVERFLOW	Table B averages for the overflow area of a sorted file.
MASTER	Table B averages for the master area of a sorted file.
pagenumber, pagenumber1, and *pagenumber2*	Specific pages in Table B.
PAGE	Table B averages for the specified file page.
PAGES	Table B averages for page *number1* through page *number2*.

Example

```
TABLEBX
8          AVG. FREE SPACE PER PAGE
220        AVG. FREE SLOTS PER PAGE
1025       NUMBER OF PAGES PROCESSED

256        BRECPPG - TABLE B RECORDS PER PAGE
17         BRESERVE - TABLE B RESERVED SPACE PER PAGE
99         AVG. FREE SPACE PER PAGE
243        AVG. FREE SLOTS PER PAGE
8652       NUMBER OF PAGES PROCESSED

256        XRECPPG - TABLE X EXTENSION SLOTS PER PAGE
17         XRESERVE - TABLE X RESERVED SPACE PER PAGE
```

Usage notes

You can issue a TABLEBX command only in file context: that is, the current default must be a file, not a group.

TABLEBX command output can be routed to a USE dataset.

If you specify a TABLEBX RECLEN *x* command, where *x* is something other than end-of-line, the following error is issued.

```
M204.0643: AN OPERAND IS INVALID OR MISSING
```

Using the TABLEBX command without any option provides only the averages over Tables B and X active pages. For a sorted file, the averages are computed separately for the overflow and master pages, and they appear separately in the output.

Using the LIST option

To get the exact amounts of free space and free slots on each active page, you must use the LIST option. This option produces a printed line for each Table B and Table X page scanned, starting with the lowest active page and ending with the highest one. The output of the LIST option, which precedes the averages shown above, is displayed in the following format:

```
PAGE NO.  FREE SPACE  FREE SLOTS

nnn       nnn         nnn
.         .           .
.         .           .
.         .           .
```

If the current file is a sorted one, each line includes an indication of the type of page that the line describes. For example, if page number 5 is an overflow page, the output might be:

```
PAGE NO.  FREE SPACE  FREE SLOTS
5         200         28        OVERFLOW
```

Limiting the range of pages scanned

You can limit the range of Table BX pages to be scanned in the following ways:

- Specify the PAGE or PAGES option to select a single page or a range of pages.

  The following command causes a scan of five pages, if they fall in the range of active pages. If any of the selected pages is inactive, an error message is issued and the command is ignored. Page numbers are specified in decimal and cannot be negative. For example:

  ```
  TABLEBX PAGES 0 TO 4
  ```

- Limit the range of pages scanned. This way applies only to sorted files. Use the options OVERFLOW or MASTER. In either case, the scan is done only over the specified pages.

Using the RECORD LENGTH option

The RECORD LENGTH option provides an estimate of the average length of a logical record in Table B. A logical record is a record that includes all its extensions.

The average record length is exact unless DELETE RECORDS or IFDSET is used in the file. If DELETE RECORDS or IFDSET is used, the record length computed is too big, because space from the deleted records is never released.

The RECORD LENGTH option is available only when all of Tables B and X are scanned. Or, to state it another way, RECLEN option does not work with either the PAGE *n* or the PAGES *n1* TO *n2* option.

TABLEE: Determining Table E use

Status A new command in V7R1.0, TABLEE, collects Table E usage statistics and finds segments where compaction will be most effective.

Purpose A command to produce Table E statistics:

Syntax `TABLEE FROM nn TO nn SEG`

Where:

- *FROM nn* specifies the starting segment number. The default is 0.
- *TO nn* specifies the ending segment number. The default is the EHIGHPG segment.
- *SEG* lists statistics by segment. The default is no.

  When the SEG option is specified, the TABLEE command prints the total number of gaps and total gap size per segment for each Table E.

Usage CCA recommends coordinating the TABLEE command with the COMPACTE command to combine the space gaps that occur over time in the extension records.

Example The following is an example of TABLEE output.

```
IN TESTE TABLEE SEG

M204.2817: SEGMENT:  0   NUMBER OF GAPS:  364   TOTAL GAP SIZE:  1820
M204.2817: SEGMENT:  1   NUMBER OF GAPS:  229   TOTAL GAP SIZE:  1144
M204.2815: TOTAL NUMBER OF GAPS: 593
M204.2816: TOTAL GAP SIZE: 2964
```

TABLEX: Determining Table X use

Status New in V7R1.0

Privileges File manager

Function Displays information about page usage in Table X of a Model 204 file

Syntax

```
TABLEX [LIST] [RECORD LENGTH]
 [PAGE pagenumber]
 [PAGES pagenumber1 TO pagenumber2]
```

Where The following options display page use in Table X. A TABLEX command accepts only one option per command, except if the LIST option is specified, which may be followed by the RECORD LENGTH option.

Option	Displays…
LIST	One line for each Table X page scanned, from the lowest to the highest page.
RECORD LENGTH	RECORD LENGTH can be abbreviated as RECLEN. If XSIZE=0, RECLEN yields the message: `M204.2768:FILE filename HAS NO TABLE X PAGES` If XSIZE>0, RECLEN yields average length of an individual record in Table X.
pagenumber, *pagenumber1*, and *pagenumber2*	Specific pages in Table X.
PAGE	Table X averages for the specified file page.
PAGES	Table X averages for *pagenumber1* through *pagenumber2*.

Example

```
TABLEX
507  AVG. FREE SPACE PER PAGE
  2  AVG. FREE SLOTS PER PAGE
 68  NUMBER OF PAGES PROCESSED
 16  XRECPPG - TABLE X RECORDS PER PAGE
 17  XRESERVE - TABLE X RESERVED SPACE PER PAGE
```

Usage notes You can issue a TABLEX command only in file context—that is, the current default must be a file, not a group.

TABLEX command output can be routed to a USE dataset.

If you specify a TABLEX RECLEN *x* command, where *x* is something other than end-of-line, the following error is issued:

```
M204.0643: AN OPERAND IS INVALID OR MISSING
```

Using the TABLEX command without any option provides only the averages (shown in the previous example) over Table X active pages.

Using the LIST option

To get the exact amounts of free space and free slots on each active page, you must use the LIST option. This option produces a printed line for each Table X page scanned, starting with the lowest active page and ending with the highest one. The output of the LIST option, which precedes the averages shown above, is displayed in the following format:

```
PAGE NO.  FREE SPACE  FREE SLOTS

nnn       nnn         nnn
.         .           .
.         .           .
.         .           .
```

If the current file is a sorted one, each line includes an indication of the type of page that the line describes. For example, if page number 5 is an overflow page, the output might be:

```
PAGE NO.  FREE SPACE  FREE SLOTS
5         200         28        OVERFLOW
```

Limiting the range of pages scanned

You can limit the range of Table X pages to be scanned in the following ways:

- Specify the PAGE or PAGES option to select a single page or a range of pages.

  The following command causes a scan of five pages, if they fall in the range of active pages. If any of the selected pages is inactive, an error message is issued and the command is ignored. Page numbers are specified in decimal and cannot be negative. For example:

  ```
  TABLEX PAGES 0 TO 4
  ```

- Limit the range of pages scanned. This way applies only to sorted files. Use the options OVERFLOW or MASTER. In either case, the scan is done only over the specified pages.

Using the RECORD LENGTH option

The RECORD LENGTH option provides an estimate of the average length of an individual (extension) record in Table X. An individual Table X record represents one extension of a Table B base record.

The average record length is exact unless DELETE RECORDS or IFDSET is used in the file. If DELETE RECORDS or IFDSET is used, the record length computed is too big, because space from the deleted records is never released.

The RECORD LENGTH option is available only when all of Table X is scanned. Or, to state it another way, RECLEN option does not work with either the PAGE *n* or the PAGES *n1* TO *n2* option.

7

New and Enhanced Messages in V7R1.0

In this chapter

- Overview
- M204 messages
- SUBSYSMGMT error messages

Overview

This chapter contains new or enhanced messages.

M204 messages

0020 message

This *message* is generic and is used mainly for informational messages. The possible messages, in bold face type, are as follows:

PROPRIETARY MESSAGES
IGCLM244 VERSION =
SUBSYSTEM NAME =

No action needed - informational only

IGCLM244 LOAD FAILED

IGCLM244 must be contained within STEPLIB, JOBLIB or LPA.

INVALID M204XSVC FOUND(linked¦xxx) - version date

The cross memory SVC must match the version of Model 204 that is running. Either an incorrect SVC was linked into Model 204 or the CCAIN parameter XMEMSVC=xxx specified an incorrect SVC number. The version and date of the found SVC are listed.

SUBSYSTEM NAME ABSENT

No subsystem name was specified within IGCLM244. Use CRAMGEN to relink IGCLM244 with an appropriate subsystem name.

CRAM SYSTEM LX FAILED

A stand-alone cross memory environment was attempted and failed. Either an LX was not available or a GETMAIN failed.

I 0 0 NOPRFIX AUDITAD

0075 INITIALIZATION COMPLETED. BUFFERS = nnnn[+nnnn]

This informational message is produced during Model 204 initialization.

This indicates the actual number of buffers allocated for this run.

If the plus sign followed by another number is present, the second number indicates the number of above the bar (at addresses greater than two gigabytes) buffers allocated.

The second number should equal the value of the NUMBUFG parameter in such a case.

I 0 0 AUDITAD

0116 BAD INPUT TO RESTORE: reason

Where *reason* can be:

- WRONG FILE NAME
- SEQ NUMBER ON PAGE IS WRONG

- PAGE NUMBER TOO LOW
- PAGE NUMBER OUT OF RANGE

The backup file is inconsistent. This is often the result of multivolume backup files being incorrectly defined in the JCL (for example, reel 2 mounted before reel 1, or reel 1 followed by an incorrect reel 2). Use of third party backup packages may also cause this message to be issued.

File manager response: The JCL for the FROM file should be examined and compared with the TO file JCL of the DUMP job.

C 20 20 AUDITER

0199 OPEN FAILED FOR VTAM APPLID%C WITH REASON CODE X'%X'

A SNA Communications Server (also known as VTAM) ACB OPEN failed. The reason code stated in the message is explained in the IBM *SNA Communications Server Macro Language Reference* manual or the *ACB/SNA Communications Server Programmer's Reference* manual. This error can be caused by:

- An incorrect name was specified in the VTAM name parameter on the User 0 CCAIN parameter line. The name must match a valid SNA Communications Server application defined in the VTAMLST file by an APPLID macro. The Model 204 default is 'M204 '.
- A value was specified in the NOTERM parameter on the User 0 parameter line that is smaller than the number of IODEV parameter cards for the type. The Model 204 initialization tries to open a second ACB with the same name as the first after it has processed NOTERM number of IODEV cards. This is an open error X'58'.

System manager response: The possible responses are:

- Verify that the VTAM name specified in your User 0 parameter line is a correct application name. Correct the parameter and rerun Model 204.
- Verify that the value for NOTERM for each set of IODEV types (7 or 37) equals the number of IODEV parameter cards. Correct the NOTERM value and rerun Model 204.
- Verify that the ACB is not already open by another application.

Operator response: Notify the system manager.

E 4 4 VIEWERR AUDITAD NOTERM OPR

0204 PARAMETER KOMMOPT OBSOLETE AND NOT RESET

0290 INDEX CHANGE ERROR, FILE=filename, FIELD=fieldname, RECNUM=X'xxxxxxxx', TYPE= X'xxxxxxxx', FIELD VALUE=X'xxxxxxxxxx'

In the process of changing the value of the field specified in the message, an inconsistency was detected between the Table B value and the index value.

File manager response: Contact CCA Customer Support and have available all information contained in the message. A reorganization of the indicated file is probably in order as soon as possible.

C 0 4 CANCEL VIEWERR AUDITER

0291 INDEX DELETE ERROR, FILE=filename, FIELD=fieldname, RECNUM=X'xxxxxxxx', TYPE=X'xxxxxxxx', FIELD VALUE=X'xxxxxxxxxx'

In the process of deleting the value of the field specified in the message, an inconsistency was detected between the Table B value and the index value.

File manager response: Contact CCA Customer Support and have available all information contained in the message. A reorganization of the indicated file is probably in order as soon as possible.

C 0 4 CANCEL VIEWERR AUDITER

0303 USER INTERRUPTED, PROCESS TERMINATED

The executing user has been interrupted by a system administrator. The process being run has been cancelled.

Response: Find out why the user was interrupted and correct the situation.

I 0 0 AUDITMS

0336 rc nn FROM CRIO op

The formats are:

Format 1: RC *nn* FROM CRIO OPEN/PUT/GET

Format 2: ECB *postcode* FROM CRFS PUT/GET

The formats are explained as follows:

- Format 1 indicates that Model 204 module, CRIO, attempted to open a channel, send a message down a channel, or receive a message from a channel. CRAM was unable to satisfy the request. The message states the return code and the operation attempted. For example, the return code 16 indicates that the CRAM channel is already open.
- Format 2 indicates that Model 204 module, CRFS, received an unexpected post code from CRAM while attempting an I/O operation on a channel. The message shows the post code and the operation attempted.

  Possible return codes include the following values:

Value	Description
4	INVALID FUNCTION
8	CB FORMAT ERROR
16	(OPEN MAST) CHANNEL ALREADY OPEN
20	(OPEN USER) NO USER LINES FREE

Value	Description
24	(OPEN) SUBTASKS NOT PROPERLY OPENED
12	(READ/WRITE) ICB NOT OPEN
16	(READ/WRITE) OTHER TASK HAS DIED
20	(READ/WRITE) UNEXPECTED COMMAND
28	NOT ENOUGH STORAGE
99	The IGCLM244 module was generated with no SVC number, indicating an SVC-less installation. The XMEMOPT X'80' bit is required for this XDM CRAM usage.
110	(VMCF WRITE) MESSAGE LIMIT EXCEEDED

System manager response: Contact CCA Customer Support and have the audit trail available.

E 4 4 AUDITAD NOTERM

0454 UNABLE TO OPEN FILE DATASET filename, ABEND CODE X'nnnn'

An error occurred while attempting to open the dataset named *filename*. The message includes the “type” which may be an ABEND CODE, a USER CODE or an ABEND/REASON, depending on the type of abend.

For instance:

- A system ABEND CODE of X'0913' indicates an OPEN subtask failure.
- A system ABEND/REASON of X'0213'/4 indicates a missing dataset.
- A USER CODE of X'0100' indicates an invalid LRECL/BLKSIZE/DSORG.

Response: Verify that the file you are attempting to open has been defined to Model 204. If your operating system has a security package such as ACF2 or RACF, check that the Model 204 Online is authorized to access the dataset. Contact your system manager if the problem persists.

System manager response: Refer to your CCAAUDIT output or console log for more specific messages related to the open failure. Correct and rerun Model 204.

If there are no further error messages to assist you, contact CCA Customer Support and have the following documentation ready:

- Audit trail
- Snap dumps

C 0 4 AUDITER

0602 FILE IS IN USE

You are executing a command or User Language request which will read or update a Model 204 file. Your attempt to access the file conflicts with the activities of other users of the file. Certain file operations require exclusive access to the file. These operations

include the INITIALIZE, REDEFINE, DELETE FIELD, INCREASE, DECREASE, DUMP, RESTORE, and RESET <file parameter> commands.

While the operations are in progress, no other user may access the file. Also, the operations cannot begin as long as other users are accessing the file. In the case of the RESET command, this message is accompanied by:

```
M204.1236: PARAMETER parameter NOT RESET
```

For the other commands, this message is accompanied by the prompt:

```
M204.1076: DO YOU REALLY WANT TO TRY AGAIN?
```

Situations involving the application subsystem in which this error can happen are as follows:

- An attempt to store a procedure which you have been editing and that the subsystem has locked.
- An attempt to access a file which conflicts with the activities of other users of the file. The subsystem uses the default response of N and exits.

Response: Respond Y to attempt to access the file again or respond N to quit.

Many commands listed above can take a long time to complete. Interactive use of the commands against large files is not recommended.

C 0 4

0783 INCORRECT DDNAME LIST

The list of ddnames entered as part of the CREATE command was specified incorrectly. This probably resulted from a CREATE file name command without the FROM keyword following the Model 204 file name. A correct example is:

```
CREATE M204FILE FROM M204FILE, M204F2
```

where the first M204FILE is the file name and the second M204FILE is the ddname. Note that FROM follows the file name.

Also, note that if you enter a trailing continuation character on the CREATE line:

```
CREATE SOMEFILE -
PARAMETER ESIZE=100
END
```

The command will be read as:

```
CREATE SOMEFILE PARAMETER ESIZE=100
END
```

In this instance, the valid keyword for the CREATE command is FROM, not PARAMETER, and the command will be rejected.

File manager response: Review the CREATE command in the *Model 204 File Manager's Guide* and retry the command.

E 0 4 AUDITER

1047 %C IS AN INVALID%C

The M204.1047 message is used for many instances where the value supplied for the specified command parameter or option is invalid.

It can appear for an invalid local ID (LOCLID)

It can appear for an invalid ECF parameter.

For the MSGCTL command, M204.1047describes various of the following errors:

%C is an invalid	Where
SNAP/DUMP OPTION	SNAP=*x* specified
CLASS	CLASS=*x* specified
RETURN CODE	RETCODEO=x or RETCODEB=x specified
PARAMETER FOR THIS MESSAGE	NOAUDITxx specified for non-xx message
OPTION	Any undefined or misspelled option specified
PREFIX	*xxxx.msgno* specified where *xxxx* is not M204,USER,SIRius
MESSAGE NUMBER	*xxxx.msgno* specified where *msgno* is not a valid message number
COMMAND	MSGCTL command is not allowed for this message number

For example, when an NOAUDITxx is entered for a message that is not of type xx, the following appears:

```
M204.1047: NOAUDITxx IS AN INVALID PARAMETER FOR THIS
MESSAGE
```

Response: Refer to the *Model 204 Parameter and Command Reference* for a list of the correct values for the parameter. Retry the command.

C 0 4 AUDITER

1069 ALLOCATE/FREE FAILED WITH [RETURN CODE return_code

| ERROR REASON CODE value

| INFORMATION REASON CODE value

| SMS REASON CODE value]

An error was detected during a dynamic allocation (SVC 99). The return and reason codes are described in the IBM manual *MVS Programming: Authorized Assembler Services Guide, GC28-1763*, Chapter 26.

Response: Correct the error and retry the command.

C 0 4

1149 parametername HAS BEEN SET TO ITS MINIMUM|MAXIMUM VALUE: value

You have attempted to reset a parameter either below the minimum or above the maximum allowed for it. Model 204 has reset the parameter to the correct minimum or maximum.

Response: Retry the command with a value within the correct range or leave the value as is since Model 204 has already reset the parameter to the correct minimum or maximum.

For PARAMETER=ESIZE, if you specify:

- A zero value, which is the equivalent of Not Applicable, no pages are allocated for Table E.
- A non zero value, then the minimum of 20 pages allocated for Table E is enforced.

E 0 0 AUDITMS (OPR)

1230 TABLE B | X FULL -- APPENDS --: FILE filename

Too many records have been added to Table B or Table X. The system has exhausted the record numbers available to the file. The file may be usable, but no more records can be added to the file table.

Response: Contact your file manager.

File manager response: Reorganize the file with more usable space in the specified table. Note that you may have a great deal of free space available on each page but are unable to use it because your slots per page are exhausted. See *Model 204 File Manager's Guide* on recommended file loading techniques and refining the parameter settings.

E 48 48 CANCEL AUDITAD

1244 BACKOUT IS INVALID FOR NON-TRANSACTION BACKOUT FILES

The user has coded the BACKOUT statement as part of a request that accesses non-transaction back out files. The files are listed in previous M204.2792 messages.

Response: Either ask your file manager to change the non-transaction back out files to transaction back out files, or do not reference the non-transaction back out files in the request that contains the BACKOUT statement.

File manager response: Alter the file options to include transaction back out, if appropriate.

C 0 4 CANCEL AUDITER

1449 ERROR WHILE PROCESSING CCASNAP

or

ERROR WHILE PROCESSSING CCASNAP, CCASNAPS DISABLED

Model 204 encountered an unexpected error while formatting a CCASNAP. Formatting of the current snap was terminated. Information about the unexpected error is audited. The system parameter SNAPFLIM specifies how many such errors are allowed before

CCSNAP formatting is terminated. The view-only parameter, SNAPFAIL, shows how many CCASNAP failures have occurred so far.

System manager response: This message indicates a problem with the CCASNAP formatter and the resulting SNAP should be saved for CCA Customer Support.

E 100 VIEWER SNAP

1856 STREAM TO BE [OFFLOADED|SWITCHED] IS NOT OPENED FOR OUTPUT, COMMAND REJECTED

The stream name, given in the OFFLOAD or SWITCH STREAM command, was not opened for output. It cannot be off loaded or switched.

Response: Verify the name of the stream you wish to off load or switch, and reissue the OFFLOAD or SWITCH STREAM command with the correct stream name.

C 0 4 AUDITER

1938 REQUEST FOR SPACE IN TABLE E EXTENDS BEYOND EHIGHPG, BIT MAP PAGES ARE SCANNED FOR FREESPACE

A User Language request attempted to store a large object and all of the append pages in Table E have been allocated. Model 204 must scan the bit map pages for previously allocated Table E pages to look for free space to store the large object.

File manager response: The file should be reorganized to compact fragmented Table E space.

I 0 0

1977 FILE%F MAY NOT BE ACCESSED REMOTELY

An attempt has been made to open a file under one of the following conditions:

- An attempt was made to remotely open a public file or a semi-public file without a valid password. This type of access is not permitted for the file unless OPENCTL has the X'08' bit on.
- An attempt was made to remotely open a file outside of APSY as part of a permanent group. This type of access is not permitted for the file unless OPENCTL has theX'04' bit on.
- An attempt was made to remotely open a private or semi-public file outside of APSY with a valid password. This type of access is not permitted for the file unless OPENCTL has the X'02' bit on.
- An attempt was made to remotely open a file that participates in Date Time Stamp processing (FOPT=X'10'). Date Time Stamp processing does not support remote file access.

File manager response: Ensure that the file has the proper OPENCTL setting.

C 0 4

2417 EMPTY RESTART STREAM ASSUMED IN FIRST PASS

This message is written when a permanent I/O error occurs while attempting to read the first record of the RESTART stream. Recovery assumes that the RESTART stream is empty. This is not an error condition if the RESTART dataset has been allocated but never used.

System manager response: Verify that the RESTART STREAM has been correctly specified as follows:

- If the DEFINE STREAM command was not specified, verify that the dataset name (DSN) specified by the RESTART DD statement of the recovery run matches the DSN associated with the CHKPOINT DD statement of the previous run.
- If the DEFINE STREAM command was specified, verify that the members specified are the same as those used during the previous run, and that the DD statements match.
- Check the JES log in the previous run to make sure that the dataset(s) was cataloged properly or if the actual physical device addresses need to be specified for the dataset(s). Use a utility to verify the existence of the dataset(s) on the specified physical device(s).

Operator response: Notify your system manager.

E 0 4 OPR

2488 (HARD | SOFT) RESTART OF USER AFTER EOJ - ATTEMPTED FILE UPDATE BLOCKED

A file update was attempted after the Online termination checkpoint was taken while an existing transaction was still active. A hard or soft restart has been done, participating files may be marked as physically or logically inconsistent, and EOJ processing continues normally.

System manager response: Send the snap to CCA Customer Support

C 0 4 NOTERM

2581 XMEMOPT=2 (IOS BRANCH) REQUIRED FOR option

The indicated *option* is only valid if IOS branch entry (the X'02' bit in the XMEMOPT CCAIN parameter) is being used. Possible options are a NUMBUFG greater than 0 or the XTIOT option.

The XTIOT option might be specified explicitly on a DEFINE or ALLOCATE command, or as the default by setting the SYSOPT2 X'80' bit. The NUMBUFG value is set in a CCAIN parameter.

Response: Either remove the XTIOT option from the DEFINE or ALLOCATE command or contact the system manager.

System manager response: If the message is issued during initialization:

- Either, remove the X'80' bit from the SYSOPT2 parameter (if the complaint is about the XTIOT option)
- Or, set NUMBUFG to 0 (if the complaint is about the NUMBUFG parameter)

- Or, restart M204 with XMEMOPT=X'02' and the appropriate SVC entry

C 0 4 AUDITMS

2605 CHKPOINT TOO SMALL FOR ROLL FORWARD - xxx BLOCKS REQUIRED; yyy FOUND

To guarantee that the ROLL FORWARD phase does not run out of space when writing to the CHKPOINT dataset, the primary space allocation for this CHKPOINT dataset must be larger than the total space allocation of the CHKPOINT dataset in the original run being recovered.

System manager response: Allocate a larger, non-GDG CHKPOINT dataset for the recovery step. GDGs are not supported for the CHKPOINT dataset in a recovery step.

E 52 52 AUDITAD OPR

2612 CHECKPOINT COMMAND UNSUCCESSFUL -%C

CHECKPOINTING NOT SPECIFIED FOR THIS RUN

| CCAIN PARM NUSERS =1

| IN SYSTEM INITIALIZATION, TERMINATION

| INVALID TRANSITION

| REDUNDANT TRANSITION SPECIFIED

| OPERATOR HAS CONTROL

The reason the previously issued command didn't complete successfully.

System manager response: Reissue in the appropriate context.

Response: Reissue in the appropriate context.

C 0 4

2629 ONEPASS DISALLOWED ACROSS FILE RECOVERY

The REGENERATE ONEPASS command was specified for a file that was recovered (ROLL BACK/ROLL FORWARD) during the regenerate time span. This message is followed by another message.

If message is …	Indicating…	Then…
M204.1436	Which file was removed from the regenerate process	The file is left in the state that it existed just prior to the recovery. Processing for all other files continues.
M204.1435	More serious errors were encountered	REGENERATE processing should not use the ONEPASS option for this file combination.

File manager response: Two passes of the journal are required to regenerate the file. Rerun the job removing the ONEPASS option. If the file was not restored as part of the REGENERATE command, it must be restored again.

C 0 4

2630 STATEMENT TOO COMPLICATED

The User Language statement that you coded consumed more than 16K (16-byte entries) of QTBL space.

Response: Split the single statement into multiple statements.

Note: This message may be issued after the following message; in that case, increase the size of QTBL and retry the request.

```
M204.0211: REQUEST TOO LONG - QTBL
```

C 0 4 AUDITER

2654 TABLE E | X DOES NOT EXIST

The specified table does not exist.

Response: Use an INCREASE TABLEE command on only files with Table E created. Use an INCREASE TABLEX command on only files with Table X created.

C 0 4 AUDITER

2677 ALL FREE PAGES ALLOWED FOR COMPACTION HAVE BEEN USED. COMMAND COMPACTB ENDS.

Compactor had used all free pages it was allowed to use.

File manager response: Use the FREE argument to increase the number of Table B pages available for use by COMPACTB processing.

C 0 4 AUDITER

2684 CHECKPOINT CONFIGURATION CONFLICT -%C%C

CHKPNTS IS OPEN SO CPTIME MUST BE NON-ZERO

| CHKPNTS IS OPEN SO DKUPDTWT MUST BE ZERO

| CHKPNTS IS OPEN SO KOMMOPT MUST BE SET TO 1

| CHKPNTS IS NOT OPEN AND CPTYPE = 1

| CHKPNTS IS NOT OPEN AND CPTS IS NON-ZERO

| {CHKPOINT | CHKPNTS} CONTAINS A PARALLEL OR RING DEFINITION

| {CHKPOINT | CHKPNTS} HAS MORE THAN 16 LEVELS OF STREAM DEFINITION RECURSION

| {CHKPOINT | CHKPNTS} CONTAINS A RING DEFINITION

Parameters associated with checkpoints are consistently checked when checkpoint streams are opened, at the beginning of Online processing and also during the processing of a RESET command. The indicated error has been found with the result that the Online either failed to come up or the RESET command was rejected.

Note: If CPTYPE=1, neither CHKPOINT nor CHKPNTS may be configured as a ring stream, a parallel stream, or a CMS formatted dataset.

System manager response: If the Online failed, change the appropriate CCAIN parameter line entries and/or the CHKPOINT or CHKPNTS stream definitions; if the RESET command failed, take appropriate action.

I 0 0 AUDITMS

2702 MISMATCH BETWEEN TABLE %C QUEUE LENGTH AFTER REBUILD AND NUMBER OF PAGES ADDED TO THE QUEUE: %C'

BLDREUSE command with option NEW found that the new queue length (BQLEN) is not equal to the number of pages added to the queue during queue rebuild.

File manager response: Run the command again. If the error persists, report it to CCA Customer Support.

C 0 4 AUDITER

2709 *LOOK CORE X"%X"X"%X" '***

In response to $UBUF, displays the *LOOK command to use to view the buffer.

I 0 0 AUDITMS

2710 ERROR %C SETTING TCP/IP SYSID %C

The TCPNAME parameter must be set to the 2-byte System ID of the z/VSE partition running TCP/IP for z/VSE.

System manager response: Check that the SYSID in the message is the same as the System ID of the TCP/IP partition.

C 0 4 AUDITER

2711 %C IS NOT A DUMP DATASET, CANNOT RESTORE INTO %C

The FROM clause of the RESTORE command points to a dataset that is not a dump dataset.

File manager response: Ensure that the FROM DD points to a dataset created by the Model 204 DUMP command. Note that a dump dataset created by V7R1.0 or later is not useable in earlier versions.

C 52 52 AUDITER

2712 STREAM %C %C %C

Informational messages are produced during SWITCH STREAM command processing. The following messages are possible:

STREAM %C IS BEING SWITCHED

A successful SWITCH command has been issued.

MEMBER %C IS EMPTY AND CANNOT BE SWITCHED

A member of the stream being switched has no records. It will not be switched in order to prevent empty members.

STREAM %C IS INELIGIBLE FOR SWITCHING

The stream cannot be switched because there are no further members behind the current one.

STREAM %C - NOT ALL MEMBERS SWITCHED

A parallel stream was switched, but not all members of the stream could be switched. This message is preceded by additional M204.2712 informational messages.

STREAM %C SWITCHED VIA COMMAND

A successful SWITCH command was issued. All datasets switched.

SWITCH %C WAS UNSUCCESSFUL

An unsuccessful switch command was issued. No dataset switched.

I 0 0 NOTERM OPR

2713 INSERT NOT SUPPORTED FOR LARGE OBJECT FIELDS

The User Language INSERT statement is not yet supported for use on Large Object field types.

Response: Remove the INSERT statement from the User Language request.

C 0 4 ECHO AUDITER

2714 %C SPECIFIED FOR BOTH DUMP DDNAME AND FILENAME

On a RESTORE, RESTOREG or REGENERATE…FROM command, the DDNAME of the dump dataset was the same as the filename being restored into. You probably mistyped the DDNAME of the dump dataset.

File manager response: Correct the command and reissue.

C 20 20 AUDITER

2715 ATTEMPT TO ACCESS DATA PAST END OF LOB

An assignment to a buffer from a Large Object attempted to copy data past the end of the Large Object into the buffer.

If this error is detected at evaluation time, the request is cancelled.

Response: Correct the length or position of Large Object reference to remain within the bounds of the object.

C 0 4 CANCEL AUDITER

2716 FREE SPACE IS EMPTY

The message indicates that there are no pages in the free space.

File manager response: Issue an INCREASE DATASET command or decrease Table B, D, E, or X to add pages to the free space.

I 0 0 AUDITAD

2717 AUTOMATIC INCREASE FOR %F TABLE %C IS NOT DONE DUE TO EMPTY FREE SPACE

The automatic increase attempted when file is opened failed due to empty free space.

File manager response: Run INCREASE DATASET command or decrease Table B, D, E, or X to add pages to the free space.

C 0 4 AUDITER

2718 SEND STATEMENT BYTES VALUE LESS THAN ONE

The SEND statement uses the optional FOR *n* BYTES syntax, but the value of *n* is zero or negative.

Response: Correct the SEND statement to use a value of *n* greater than zero for the number of bytes to send.

C 0 4 ECHO AUDITER

2719 OPEN %C FAILED. FILE UPDATE IN PROGRESS

An OPEN or OPENC was issued at command level for a file that has a transaction in progress due to DKUPDTWT set > 0.

Response: Wait until DKUPDTWT has expired before reopening the file.

C 0 4 AUDITER

2720 VIO NOT ALLOWED WITH XMEMOPT=2

An attempt was made to open a file whose DD card indicates a VIO dataset, but the IOS Branch Entry feature is enabled by the XMEMOPT=2 bit. When IOS Branch Entry is being used, VIO datasets may not be used as Model 204 files.

System manager response: Probably the use of the VIO dataset should be eliminated. If VIO was being used for CCATEMP, it is much better to use CCATEMP in storage by setting TEMPPAGE. If there is a compelling reason to use a VIO dataset, then the XMEMOPT=2 bit should not be set, though this will adversely affect I/O CPU overhead and prevent the use of above-the-bar buffers (NUMBUFG>0).

C 0 4 AUDITER

2721 TCP ADDRESS %C ALREADY USES PORT %C ON INTERFACE %C

The $SOCKET INIT call is being used to define a new TCP/IP interface, but there is already an active interface with the same address and port. The new interface is not created, and $STATUSD is set to -999.

Response: Change the new interface to use a different port.

E 0 4 AUDITER

2722 GROUP CLOSED: groupname

The named group has been successfully closed.

I 0 0 AUDITMS

2723 FOPT=X'10' INVALID WHEN DTSFN NOT SPECIFIED IN RUN

You may not enable date/time stamp processing by setting FOPT to include X'10' for a file when the DTSFN parameter has not been specified for a run.

File manager response: Do not set FOPT=X'10' for a file when the DTSFN parameter has not been provided.

System manager response: Include the DTSFN parameter in your job stream when you require date/time processing.

E 0 4 AUDITMS

2724 PARAMETER “DELETE” IS INDICATED FOR %F WITH NO LPU FEATURE ENABLED AND NOT A SINGLE USER RUN. AN EXCLUSIVE FILE ENQUEUE WILL BE REQUIRED.

Response: Parameter “DELETE” has been used to compact a file with lock pending updates feature disabled and the run is not single user (NUSERS=1). If exclusive file enqueue will not be available, the compactor will not run.

I 0 0 NOTERM OPR

2725 FILE OPTIONS X’10’ AND X’02’ ARE INCOMPATIBLE

You attempted to enable date/time stamp processing (FOPT=’10’) at the same time you disabled lock pending updates (FOPT=X’02’) for a file. You cannot specify both of these file options simultaneously. Files that participate in date/time stamp processing must have lock pending updates enabled.

File manager response: Either remove the FOPT=X’02’ setting or the FOPT=X’10’.

E 0 4 AUDITMS

2726 PREVIOUS COMPACTION FOR FILE filename HAS NOT FINISHED

Another compaction of the indicated file is in progress.

File manager response: Only one compaction of a file allowed at any time. Run the compactor again when the previous compaction is finished.

C 0 4 AUDITER

2727 CAN’T DELETE DTS FIELD WHEN FOPT=X’10’ IS ON

Date/time stamp files require the presence of the DTSFN field.

File manager response: If this file participates in date/time stamp processing, then do not delete or rename this required field. If you want to remove support for the date/time stamp feature from this file, first reset the FOPT parameter, removing the X’10’ setting. Then you can delete or rename the field.

E 0 4 AUDITMS

2728 DATE TIME STAMP FIELD MAY NOT BE INVISIBLE, UNIQUE, NOR HAVE MULTIPLE OCCURRENCES

You attempted to activate the date/time stamp feature by setting the FOPT parameter to include X’10’, but the DTSFN field for the file was defined with an incompatible attribute: INVISIBLE, UNIQUE, and/or OCCURS greater than 1, which are invalid field types for the date/time stamp field.

File manager response: Redefine the DTFSN field so that it does not specify the INVISIBLE, UNIQUE, or OCCURS *n* keywords. You may include OCCURS=1.

E 0 4 AUDITMS

2729 DATE TIME STAMP FILE MAY NOT BE OPENED WHEN DTSFN NOT SPECIFIED IN RUN

You cannot open a file that includes FOPT=X’10’ when the job does not support the date/time stamp feature.

File manager response: To process a date/time stamp file in a run that does not support date/time stamp processing, exclude FOPT value X’10’. The date/time stamp field will not be automatically updated. Remember to reactivate date/time stamp processing with the FOPT value X’10’ for later runs that include the DTSFN parameter.

System manager response: Provide the DTSFN parameter in the job stream when DTS processing is desired.

E 0 4 AUDITMS

2730 DATE TIME STAMP FILE, BUT DTSFN NOT DEFINED IN FILE

The FOPT=X'10' bit is set for this file, but the field, as specified in the DTSFN system parameter, does not exist in the file.

There are three approaches for resolving this issue:

1. Change the DTSFN parameter for your run to match one of the date/time stamp fields in the file
2. Define a new field in the file that matches the current value of the DTSFN parameter in the new run
3. Remove the FOPT=X'10' setting from the file, and process the file without date/time stamp support

E 0 4 AUDITMS

2731 DEBUGOPT SPECIFIED, BUT XMEMOPT NOT SET

To utilize the server swapping features of DEBUGOPT, the cross memory environment must be initialized in the Online.

System manager response: Set up the cross memory environment in the Online by linking in the M204XSVC module or by setting the XMEMSVC parameter along with XMEMOPT=X'04'

C 80 80

2732 %F HAS NO TABLE X DEFINED. PARAMETER “DELETE” NOT ALLOWED.

The parameter “DELETE” has been used with the file that has no Table X defined. The command is rejected.

File manager response: Do not use parameter “DELETE” on files with no Table X defined.

C 0 4 AUDITER

2733 DTS OPTION IS NOT ALLOWED FOR FILES CREATED BEFORE RELEASE 6.1

The DTS feature is supported only in files created by Model 204 V6R1.0 or later. You cannot, therefore, set FOPT=X'10' in files created by an earlier release.

File manager response: Use the DTS option only in files created by V6R1.0 or later.

E 0 4 AUDITMS

2734 IARV service ERROR - RETURN CODE=retcode, REASON CODE=reascode

The IARV64 z/OS system service failed with the indicated error. IARV64 system services are required when above the bar storage is requested, most likely as the result of the setting of the NUMBUFG CCAIN parameter, which requests above the bar disk buffers.

System manager response: Consult the IARV64 documentation in the *"Authorized Assembler Services Reference"* for the release of z/OS being run.

In the unlikely event that this is not helpful (since this error should not occur in normal use of any above the bar storage facilities), contact CCA Customer Support.

E 0 4 AUDITMS

2735 %F HAS TOO MANY RECORDS PER TABLE B PAGE TO SATISFY RECORD LIMIT REQUIREMENT

Parameter BRECPPG is too large to be used with RECRDOPT=1

Response: Lower BRECPPG parameter or remove RECRDOPT=1

C 0 4 AUDITER

2736 FILE filename FOPT=X'10' TURNED OFF: DATE TIME STAMP FIELD UNDEFINED

You issued the INITIALIZE command for the file *filename* with date/time stamp updates. Since the KEEPDEFS option was not specified, INITIALIZE processing removed field definitions. This disabled the date/time stamp feature (FOPT=X'10').

File manager response: Define the date/time stamp field and reset FOPT to include X'10'.

I 0 0 AUDITAD

2737 CHKPOINT DATASET MUST BE COPIED TO LARGER DATASET - DO NOT DELETE AND REALLOCATE!

During a secondary recovery, the space in the CHKPOINT dataset has been exhausted. A new, larger dataset must be allocated. However, the contents of the original dataset must not be lost. The original CHKPOINT dataset has recovery information that is required for the ROLL BACK recovery to succeed. Copy the contents of the old dataset into the new, larger dataset before rerunning recovery.

IF YOU DELETE THE DATASET, YOU WILL NOT BE ABLE TO RECOVER!

System manager response: A new, larger CHKPOINT dataset must be allocated and the contents of the old CHKPOINT dataset must be copied to the new, expanded dataset.

E 0 8 NOTERM OPR

2738 COMMAND REQUIRES FILE MANAGER PRIVILEGE FOR FILE filename

You issued a Model 204 command which requires file manager privileges for the specified file. You do not have these privileges as determined by your file password or the default privileges of the indicated file.

Response: Have your file manager perform the function.

File manager response: Open the file with the password required for file manager privileges.

E 0 4 AUDITMS

2739 PARAMETER parm CANNOT BE RESET ON USER 0'S PARAMETER LINE: %C

You set parameter *parm* on the User 0 parameter line, but the parameter cannot be set there. The value is ignored, the default value for this parameter remains in effect.

System manager response: Remove this parameter from the User 0 parameters, since it will not change the value of the parameter.

E 0 4 AUDITAD

2740 INVALID PARAMETER parm: feature SUPPORT NOT LINKED IN

You have specified the *parm* parameter on the User 0 parameter line, but the *feature* which supports this parameter is not linked into the load module.

System manager response: Remove the *parm* parameter from the User 0 parameters, or link the required object decks that support *feature* into the load module.

E 0 4 AUDITAD

2741 ROLLBACK/ROLL FORWARD MUST BE RUN PRIOR TO RERUNRB - RESTART CANCELLED

RESTART RERUNRB has been specified to repeat roll back for a previous recovery run. However, there was no previous recovery executed. RERUNRB can only be used following a previously successful recovery run, either ROLL BACK or ROLL BACK ROLL FORWARD. RERUNRB may be required when the previous recovery run succeeded but one (or more) file(s) needing recovery was missing from the JCL.

System manager response: Submit the recovery run again, specifying either RESTART ROLL BACK or RESTART ROLL BACK ROLL FORWARD.

E 0 8 NOTERM

2742 BUG .. NEW RECORD NUMBER NOT BASIC FOR FILE filename

A Model 204 system error has occurred while restoring a file compacted with the COMPACTB command.

System manager response: Contact CCA Customer Support. Have the following documentation available:

- Audit trail from the previous run
- Snap
- Output from the CCARF dataset

Run secondary recovery using the CHKPOINT dataset as the RESTART dataset and omit the failing file from the JCL. If the failing file was dynamically allocated, rename the dataset to cause the open to fail.

E 0 4 DUMP SNAP AUDITAD

2743 BUG .. FIRST RECORD NOT BASIC FOR FILE filename

A Model 204 system error has occurred while restoring a file compacted with the COMPACTB command.

System manager response: Contact CCA Customer Support. Have the following documentation available:

- Audit trail from the previous run
- Snap
- output from the CCARF dataset

Run secondary recovery using the CHKPOINT dataset as the RESTART dataset and omit the failing file from the JCL. If the failing file was dynamically allocated, rename the dataset to cause the open to fail.

E 0 4 DUMP SNAP AUDITAD

2744 BUG .. NOT ENOUGH SPACE ON PAGE FOR FILE filename

A Model 204 system error has occurred while restoring a file compacted with the COMPACTB command.

System manager response: Contact CCA Customer Support. Have the following documentation available:

- Audit trail from the previous run
- Snap
- Output from the CCARF dataset

Run secondary recovery using the CHKPOINT dataset as the RESTART dataset and omit the failing file from the JCL. If the failing file was dynamically allocated, rename the dataset to cause the open to fail.

E 0 4 DUMP SNAP AUDITAD

2745 BUG .. EXTENSION POINTER NOT ZERO FOR FILE filename

A Model 204 system error has occurred while restoring a file compacted with the COMPACTB command.

System manager response: Contact CCA Customer Support. Have the following documentation available:

- Audit trail from the previous run
- Snap
- Output from the CCARF dataset

Run secondary recovery using the CHKPOINT dataset as the RESTART dataset and omit the failing file from the JCL. If the failing file was dynamically allocated, rename the dataset to cause the open to fail.

E 0 4 DUMP SNAP AUDITAD

2746 XSIZE*XRECPPG MUST NOT EXCEED 536870912

The product of XSIZE and XRECPPG must not exceed 16777216 times 32 or 536870912. This specifies the maximum number of extension records allowed in a single Model 204 online when using Table X. Also, setting both XRECPPG and XSIZE to 1 is invalid. The CREATE command is rejected.

File manager response: Retry the command specifying values for XSIZE and XRECPPG whose product does not exceed the 526870911 limit. Refer to the *Model 204 File Manager's Guide* for more information on file sizes.

C 0 4

2747 TABLE X NOT SUPPORTED UNDER SORTED OR HASHED ORGANIZATION

Table X for extension records cannot be specified for files with FILEORG=8 (hashed) or FILEORG=1 (sorted). The create is rejected.

File manager response: Either remove the XSIZE parameter or specify XSIZE=0 or change the FILEORG parameter to specify a different organization.

C 0 4 AUDITER

2748 STARTING RECORD NUMBER IS TOO BIG

Starting record number indicated for COMPACTB command is beyond the range of records in the file

File manager response: Correct starting record number

C 0 4

2749 NUMBER OF BASIC RECORDS PROCESSED: %C

Informational message indicating final statistics for COMPACTB command

I 0 0

2750 NUMBER OF EXTENSION RECORDS BEFORE COMPACTION: %C

Informational message indicating final statistics for COMPACTB command

I 0 0

2751 NUMBER OF EXTENSION RECORDS AFTER COMPACTION: %C

Informational message indicating final statistics for COMPACTB command

I 0 0 AUDITMS

2752 NUMBER OF NOT PROCESSED (LOCKED) RECORDS: %C

Informational message indicating final statistics for COMPACTB command

I 0 0 AUDITMS

2753 NUMBER OF FREE PAGES USED: %C

Informational message indicating final statistics for COMPACTB command

I 0 0

2754 NUMBER OF DELETED LOGICALLY DELETED RECORDS: *count*

Informational message indicating final statistics for COMPACTB command

I 0 0

2755 UNEXPECTED ERROR DURING PHYSICAL DELETE. COMPACTION ENDS.

An error happened during COMPACTB command with the DELETE option.

File manager response: Contact CCA Customer Support and provide the required documentation.

C O 4

2756 NUMBUFG RESET TO NLRUQG * ((NSERVS + NSUBTKS) * MAXOBUF + 15 = nqueues * ((nservs + maxpsts) * maxobuf + 15) = value

If NUMBUFG is nonzero, its minimum valid value is the number of servers (the value of the NSERVS parameter) plus the maximum number of PSTs (the value of the NSUBTKS parameter) times the value of the MAXOBUF parameter plus fifteen, all multiplied by the number of disk buffer LRU queues (the value of the NLRUQG parameter). This message is generated because a smaller value was specified for NUMBUFG.

Note that if NLRUQG is not set explicitly, it is set to the same value as NLRUQ. The value of NUMBUFG has been automatically reset to its minimum allowable value as indicated by this message.

System manager response: This is an informational message and no action is required, though it is generally a bad idea to be running with such a small number of above the bar buffers relative to the number of servers.

In fact, with so few above the bar buffers, there is probably little point in using above the bar buffers, at all.

I 0 0

2757 INSUFFICIENT ABOVE THE BAR STORAGE -- RUN TERMINATED

The amount of above the bar storage required by the Online exceeds either the explicit MEMLIMIT specified on the job step EXEC card, or the default MEMLIMIT set for the entire system.

Under CMS, this message is issued if the required above the bar storage exceeds the Online's virtual machine size minus two gigabytes, the first two gigabytes of storage being below the bar.

System manager response: Either decrease the value of the NUMBUFG parameter in the CCAIN stream, since the biggest user of above the bar storage is above the bar disk buffers, or increase the amount of above the bar storage available to the Online.

Under z/OS, the latter involves increasing the explicit or implicit MEMLIMIT for the job. (The z/OS system programmer might need to get involved in increasing any system-wide limits.)

Under CMS, this would involve increasing the virtual machine size with a CP DEFINE STORAGE command. (The z/VM system manager might need to get involved in increasing the user ID's virtual storage limit in the user directory.)

E 80 80 OPR

2758 MAXBUF + NUMBUFG CAN'T EXCEED 2G

The sum of the MAXBUF and NUMBUFG CCAIN parameters exceeds two gigabytes.

This message should never occur because the maximum value of MAXBUF is 300000 (300 thousand) and of NUMBUFG is 700000000 (700 million), so the maximum possible value for MAXBUF + NUMBUFG is 700300000 (700 million, 300 thousand), which is still less than two gigabytes.

Note that these numbers indicate the number of buffers, not the number of bytes, so they represent huge quantities of virtual storage. For example, the NUMBUFG limit represents over 4.2 terabytes of virtual storage.

System manager response: Contact CCA Customer Support

E 0 4 OPR

2759 %C NOT SUPPORTED UNDER CMS

No response is adequate.

E 0 4

2760 %C NOT ALLOWED WITH NUMBUFG > 0

A command was issued which is not allowed when the NUMBUFG parameter is set to a nonzero value.

File manager response: Reissue the request in an Online with NUMBUFG set to 0, the default.

E 0 4

2761 %C DETECTED - BACKOUT WILL FOLLOW

One of the following conditions was raised by the system during processing of Date Time Stamp updates:

- CANCEL REQUEST AND INCLUDES
- USER CONNECTION LOST (PHONE WAS HUNG UP)
- CANCEL REQUEST/COMMAND IMMEDIATELY
- USER WAS BUMPED

The Date Time Stamp updates are abandoned, and the transaction is backed out.

I 0 0 AUDITAD

2762 %F HAS NO TABLEX DEFINED. PARAMETER “TABLEX” MAY NOT BE USED

The command BLDREUSE with parameter TABLEX was issued for a file with no Table X defined. The command is rejected.

File manager response: Remove parameter TABLEX.

C 0 4

2763 TOTAL NUMBER OF TABLE %C PAGES PROCESSED: %C

Informational message indicating final statistics for BLDREUSE command.

I 0 0 AUDITMS

2764 NUMBER OF TABLE %C PAGES ADDED TO THE REUSE QUEUE: %C

Informational message indicating final statistics for BLDREUSE command.

I 0 0 AUDITMS

2765 TABLE %C REUSE QUEUE LENGTH BEFORE REBUILD: %C

Informational message indicating final statistics for BLDREUSE command.

I 0 0 AUDITMS

2766 TABLE %C NUMBER OF PAGES THAT WERE ON THE REUSE QUEUE: %C

Informational message indicating final statistics for BLDREUSE command.

I 0 0

2767 TABLE %C REUSE QUEUE LENGTH AFTER REBUILD: %C

Informational message indicating final statistics for BLDREUSE command.

I 0 0

2768 %F HAS NO TABLE X PAGES

Informational message only.

I 0 0

2769 %F HAS NO TABLEX PAGES IN USE

This is an informational message.

I 0 0

2770 %F PARAMETER RECRDOPT=1 REQUIRES TABLE X TO BE DEFINED

Parameter RECRDOPT=1 may only be used when Table X is defined.

File manager response: Do not use parameter RECRDOPT=1 for files with no Table X defined.

C 0 4

2771 ATTEMPT TO UPDATE TBO AND NON-TBO FILES IN THE SAME TRANSACTION

C 0 4 CANCEL

2772 FILE ALREADY AT DYNAMIC MAXIMUM

An INCREASE TABLEB command was issued with the DYNAMIC parameter, but the file was already at the maximum Table B size for which space was reserved by the MAXINCBP system parameter.

File manager response: The INCREASE TABLEB command must be issued without the DYNAMIC option to increase the file's Table B size. This might not be possible until there are no users running requests against the file, and there are no more pre-compiled APSY requests against the file. This latter requirement probably means that all subsystems that reference the file must be stopped.

E 0 4 AUDITER

2773 THE LPM WILL BE TOO BIG

The Logical Page Map (LPM) after INCREASE TABLE or INCREASE DATASET command will be too big and will not fit the FPL page

File manager response: Reorganize the file to make LPM smaller

C 0 4

2774 ANOTHER INCREASE COMMAND IS IN PROGRESS

An INCREASE COMMAND or auto increase takes place while another increase command has not finished yet.

File manager response: Do not issue more than one increase command at a time.

C 0 4

2775 INVALID EXPRESSION IN INITIAL CLAUSE

The expression that is specified in the INITIAL clause is invalid.

Response: Check to make sure that the INITIAL clause does not reference any of the following:

1. Undefined %variable
2. Image item
3. Screen item
4. Field name
5. Subroutine parameter

C 0 4 ECHO

2776 %C SCHEDULER ABEND, MODEL 204 TERMINATING

This message is issued if Model 204 encounters a severe error while in either the main task or subtask scheduler. Such an error causes the run to terminate immediately.

System manager response: Call CCA Customer Support

E 0 4 AUDITAD NOTERM OPR

2777 parm1name PARAMETER RESET TO LARGER parm2name

A parameter has been reset to the value of a corresponding parameter.

System manager response: Set the value of the first parameter in CCAIN to a value equal to or greater than the value of the second parameter.

I 0 0 AUDITAD OPR

2778 DB2THRD NOT ZERO BUT MODULE ULDB IS NOT LINKED.

Parameter DB2THRD is not zero indicating DB2 interface is requested, but module ULDB is not linked in.

System manager response: Relink Model 204 with ULDB.

C 0 4

2779 UNDECLARED VARIABLE IN INITIAL CLAUSE

An undeclared %variable was used inside an INITIAL clause. All % variables used inside an INITIAL clause must be declared, even if there is a default type for undeclared %variables.

Response: Declare the %variable used in the INITIAL clause or correct the name used in the INITIAL clause to that of a declared %variable.

C 0 4 ECHO

2780 ABEND DURING INITIALIZATION, MODEL 204 TERMINATING

E 0 4 AUDITAD NOTERM OPR

2781 GSTATUS MUST BE G, L, SPACES, OR NULL

The GSTATUS parameter was specified with an incorrect value. The allowed values are 'L' for last message in group; 'G', for other messages in a group; and blank or NULL, for messages that are not part of a group.

Response: Correct the MQ statement in error.

C 0 4 ECHO AUDITER

2782 XSIZE MUST BE SPECIFIED

File manager response: The FILE CREATE command is rejected. Reissue specifying XSIZE to create a file with a Table X, or eliminate all Table X parameters to create a file without a Table X.

E 0 4

2783 TEMP GROUP DOES NOT MATCH SAVED COMPILATION, RECOMPILING

The definition of the temporary group has changed and is not compatible with the saved compilation for the pre-compiled request. The precompiled procedure included for this user will be recompiled and the new compilation will replace the previously saved compilation.

File manager response: If this message appears often then it may indicate that the procedure involved should not be precompiled. Its purpose is to highlight a possible performance issue, not to indicate a processing error.

E 0 4

2784 IGNORING COMMANDS, UNTIL BEGIN FOUND

This message is issued when LECHO=X'04' (echo lines from included procedures) and is proceeded by any of the following messages:

```
M204.2783 TEMP GROUP DOES NOT MATCH SAVED COMPILATION,
RECOMPILING
```

```
M204.2305 NODE REFERENCES DO NOT MATCH SAVED COMPILATION
REFERENCES, RECOMPILING
```

```
M204.1966 NON-SUBSYSTEM FILE REFERENCE BY TEMP GROUP,
RECOMPILING
```

Response: If a precompiled procedure needs to be recompiled for any of the above reasons, the commands that are found before the BEGIN are skipped and not rerun. This message is an informational message only and does not indicate a processing error.

I 0 0 NOTERM

2785 REWIND INVALID WITH STREAM CONFIGURATION

A rewind was attempted on a stream that is concatenated or contains a concatenated stream definition. In order for this rewind to succeed all concatenated members must have been defined as:

1. Data sets, or
2. Parallel streams whose members are all data sets.

The rewind failed because these conditions were not met.

Response: Message M204.1845 should follow this message and give information about the stream name. If the rewind was caused because CPMAX was exceeded on a CHECKPOINT/CHKPNTS stream then:

1. Either, increase CPMAX sufficiently so the rewind never occurs, or
2. Change the stream definition to conform to the above requirement.

If the stream is not a checkpoint stream then the stream definition must be changed. Note: for sub transaction enabled runs CPMAX is forced to be 1, therefore option (1) is not available in that case.

I 0 0

2786 CPQZSECS TIMER REFRESHED - EXPIRES IN %B SECONDS

1. The system is in an extended quiesce, CPQZSECS seconds have expired and CPQZACTN = X'80',or
2. the system is in an extended quiesce with CPQZSECS > 0 and a SWITCH STREAM CCAJRNL was issued. In either case the extended quiesce timer will be refreshed to expire in CPQZSECS.

1 0 0 OPR NOTERM

2787 OUTPUT INCOMPLETE: DD LINES OUTPUT = %C

A USE command has failed due to an I/O error on the output USE dataset. This could be caused by running out of space in the output dataset and receiving a B37, D37, or E37 abend. The request is cancelled and return code 24 is set in both BATCH204 and ONLINE.

Response: Increase the size of the USE dataset.

C 24 24 SUFFIX SAVE LAST AUDITER

2788 IGNORING COMMANDS, UNTIL BEGIN FOUND

This message is issued when LAUDIT=x'04'(Write LP lines, lines included from procedures) and is proceeded by any of the following messages:

```
M204.2783 TEMP GROUP DOES NOT MATCH SAVED COMPILATION,
RECOMPILING
M204.2305 NODE REFERENCES DO NOT MATCH SAVED COMPILATION
REFERENCES, RECOMPILING
M204.1966 NON-SUBSYSTEM FILE REFERENCE BY TEMP GROUP,
RECOMPILING
```

Response: If a pre-compiled procedure needs to be recompiled for any of the above reasons, the commands that are found before the BEGIN are skipped and not re-executed. This message is an informational message only and does not indicate a processing error.

I 0 0 NOAUDIT

2789 SMFSVC = nnn INVALID: SMF ACCOUNTING DISABLED

The SMFSVC number is invalid. IBM requires that all user SVCs be greater than 199 and less than 256. No SMF records will be written.

System manager response: Change the SMFSVC to a number greater than 199 and less than 256.

I 0 0 OPR

2790 INITIAL CLAUSE REFERENCES UNINITIALIZED VARIABLE: %C

The expression in an INITIAL clause contains an uninitalized variable.

Response: This message is for informational purposes only. The default value, (NULL for string, 0 for FIXED and FLOAT) is assigned to the uninitalized variable. The statement is compiled and evaluated without errors.

I 0 0 ECHO

2791 INITIAL CLAUSE ALREADY SPECIFIED IN PREVIOUS COMMON DECLARATION

INITIAL clause should only be specified on the first declaration of the COMMON variable.

Response: Eliminate INITIAL clause on subsequent declarations.

I 0 0 ECHO

2792 filename IS DEFINED AS [TBO | NON-TBO]

This message appears once for each file which violates the transaction back out rules for TBO and NON-TBO files. This message precedes either the M204.1244 or M204.2771 message.

Response: Either ask your file manager to change the non-transaction back out files to transaction back out files, or do not reference the non-transaction back out files if the request contains a BACKOUT statement.

File manager response: Alter the file options to include transaction back out, if appropriate.

E 0 4

2793 INITIAL MUST BE SPECIFIED IN FIRST COMMON DECLARATION

The INITIAL clause must be specified in the first COMMON declaration of the variable.

Response: Code INITIAL clause on the first declaration of the variable.

E 0 4 ECHO AUDITER

2794 JOURNAL BLOCK DISCREPANCY. EXPECTED BLOCK NUMBER IS %C, CURRENT BLOCK NUMBER IS %C

Informational message indicating that the fast recovery is not possible.

Response: If recovery failed submit this information to CCA.

I 0 0

2795 RESERVE KEYWORD EXPECTED

The keyword RESERVE must be specified when assigning a RESERVE value to a Large Object field.

Response: Specify RESERVE before reserve value in assignment statement for large object field.

C 0 4 ECHO

2796 SMTP %C

There was an error message returned from the SMTP email server, which is shown in this message.

Response: There is probably a syntax error in the list of addressees. Check the error message and correct the parameters to the $SNDMAIL function.

E 0 4

2797 EXPECTED COMMA - FOUND %W

The compiler parser was looking for a comma. Instead of a comma, the compiler found another word.

Response: Correct syntax and retry

C 0 4 ECHO

2798 BLOB/CLOB FIELD OR FNV REQUIRED

Syntax required a Binary or Character Large Object field, or field name variable. It was not found.

Response: Correct syntax and retry your request.

C 0 4 ECHO

2799 $SNDMAIL ERROR %C

An error has occurred while executing the $SNDMAIL function.

Response: Check the error text. In most cases the error will be due to an error in the parameters. If this is a GETHOSTBYNAME error, check that the supplied client and server names are correct.

E 0 4

2800 %L>>%L<<%L

This is the echo of a line that caused a compilation error or warning. The last token read at the time of the error (and so likely the cause of the error) is between the ">>" and "<<". This message is always preceded by a message indicating the specific nature of the error.

Response: Correct the error in the User Language program.

E 0 4 AUDITER NOPREFIX

2801 %L%L%L

E 0 4 ER NOPREFIX '

2802 '%L%L%L'

E 0 4 ER NOPREFIX

2803 '(FILE = %L, PROCEDURE = %C, LIN

E 0 4 ER NOPREFIX

2804 '%L>>%L<<%L'

I 0 0 NOPREFIX

2805 '%L%L%L'

I 0 0 NOPREFIX

2806 '%L%L%L'

I 0 0 NOPREFIX

2807 '(FILE = %L, PROCEDURE = %C, LINE = %C)'

I 0 0 NOPREFIX

2808 'BUG .. DKBM: SICK BITMAP',FFFF00

E 0 4 AD,DUMP,SNAP,NOTERM

2809 WRONG SEGMENT %C FOR PAGE %C'

Table E bitmap page has an incorrect number.

File manager response: File must be regenerated

E 0 4

2810 DESCRIPTOR NOT FOUND FOR OBJECT AT %C RECORD NUMBER %C

A discrepancy is found between object in Table E and Table B record that points to the object

File manager response: File must be regenerated

E 0 4

2811 NUMBER OF MOVED OBJECTS: %C

Informational message for compactor’s final statistics.

I 0 0'

2812 NUMBER OF MOVED PAGES: %C

Informational message for compactor’s final statistics.

I 0 0

2813 NUMBER OF RECORD LOCKING CONFLICTS: %C

Informational message for compactor’s final statistics.

I 0 0

2814 NUMBER OF MULTISEGMENT OBJECTS: %C

Informational message for compactor’s final statistics.

I 0 0'

2815 TOTAL NUMBER OF GAPS: %C

Informational message for compactor’s final statistics.

I 0 0'

2816 'TOTAL GAP SIZE: %C'

Informational message for TABLEE command indicating total gap size in pages.

I 0 0

2817 'SEGMENT: %C NUMBER OF GAPS: %C TOTAL GAP SIZE: %C'

Informational message for TABLEE command. Gap size is measured in pages.

I 0 0

2818 RECORD %C HAS NO FIELD POINTING TO OBJECT AT %C

A discrepancy is found between Table E object and Table B record pointing to the object

File manager response: File must be regenerated.

E 0 4

2819 RECORD %C NOT FOUND FOR OBJECT AT %C

A record in Table B pointing to the object is not found.

File manager response: File must be regenerated.

E 0 4

2820 DISCREPANCY FOUND FOR %C PAGE %C

A discrepancy is found between Table E object and Table B record pointing to the object

File manager response: File must be regenerated.

E 0 4

2821 OBJECT AT %B STRETCHES PAST EHIGHPG

A discrepancy is found in Table E with incorrect EHIGHPG.

File manager response: File must be regenerated.

E 0 4

2822 COMPACTION ENDS ABNORMALLY. FILE %F HAS TO BE RESTORED

A discrepancy is found in Table E.

File manager response: File needs to be recovered or regenerated depending on the preceding message

E 0 4

2823 BUG %C DURING OBJECT MOVE

A problem happened while compacting an object.

File manager response: A recovery needs to be run to restore the file

E 0 4 AD SNAP

2824 BUG .. DKFPE: NO OBJECT FOUND

Large object is about to be deleted but the object's first page does not have a proper page ID.

File manager response: If the problem happens during recovery then the file must be regenerated, otherwise a file's recovery is needed. If the problem persists, media recovery is required.

E 0 4 AD DUMP SNAP NOTERM

2825 COMPACTB AND COMPACTE MAY NOT BE RUN AT THE SAME TIME

File manager response: Run only one compactor at time.

I 0 0

2826 '%F VERSION IS PRE 6.3. COMPACTION NOT ALLOWED

An attempt to compact a file created prior to version 6.3.

File manager response: File needs to reorganized using Version 6.3.

C 0 4

2827 BUG .. DKFPE: DIFFERENT RECORD NUMBER

Large object is about to be deleted but the object's header indicates record number different from the one that contains this object's descriptor

File manager response: If problem happens during recovery then file needs to be regenerated otherwise a file's recovery is needed. If problem persists media recovery is required.

E 0 4 AD DUMP SNAP NOTERM

2828 BUG .. DKFPE: DIFFERENT ATTRIBUTE

Large object is about to be deleted but the object's header has a field attribute different than the one that has object's descriptor

File manager response: If problem happens during recovery then file needs to be regenerated otherwise a file's recovery is needed. If problem persists media recovery is required.

E 0 4 AD DUMP SNAP NOTERM

2829 REORGANIZE %F TO IMPROVE EFFICIENCY FOR TABLE E ALLOCATION

The Table E append pages have all been allocated. It is recommended that the user reorganize the file in order to improve efficiency when storing new Large Object (LOB) data.

File manager response: Reorganize the file.

I 0 0

2830 UNABLE TO ALLOCATE SDASDENQ WORK AREA - MORE ENTRIES EXIST

The SDASDENQ work area is allocated as part of ENQCTL command processing and as part of shared DASD enqueue processing. If there is insufficient storage (6K) available to allocate this work area, the ENQCTL command fails.

Response: Notify the system manager that the amount of virtual storage available to Model 204 needs to be increased.

System manager response: Increase the amount of virtual storage available to Model 204 via the REGION parameter on the job card (OS), DEFINE STOR (CMS) or partition size (VSE).

E 4 4

2831 IFAM DOES NOT SUPPORT DTS FILES

DTS files can not be used under IFAM.

Response: Do not attempt to open a DTS file.

E 0 4

2832 USE PROC ACTIVE FOR PROCEDURE

The user is attempting to edit into a procedure that is currently in use by the USE PROC command.

Response: Use a different procedure in the USE PROC command or edit a different procedure.

I 0 0

2834 ADDRESS LOOKUP FAILED FOR %C

You have specified a domain name, but TCP/IP has been unable to convert this to an internet address.

Response: Check that the name you entered is correct and known to your TCP/IP name server.

I 0 0

2835 %F TABLE %C INCREASED BY %C PAGES AUTOMATICALLY

Informational message indicating that a table was automatically increased when it became full

No response required

I 0 0 NOTERM

2836 PARAMETER name INVALID - RESET TO value

This is an informational message only.

A parameter was specified with an incorrect value. It has been reset to an internally calculated value which will accommodate other parameter settings. See the description of the parameter named to see what that calculation involves.

I 0 0

2837 NO LOGICAL CONNECTION BETWEEN PROCESS %C AND REMOTEID %C

There is an incoming request to start the named process, but the process is not one of the processes connected by definitions to the named remote ID. It could be that an incoming request to start the named process is arriving over an unexpected session.

Response: Check the definitions associated with the process, and also that the client application is using the correct VTAM LU name.

E 0 4 LAST SAVE NOTERM

SUBSYSMGMT error messages

SUM093 PFkey or command only valid with MODIFY option.

On the main SUBSYSMGMT screen, the user hit PF7/CMDprv and selected an option other than MODIFY. MODIFY is the only valid option for PF7/CMDprv.

SUM094 Subsystem name or pattern is required.

On the main SUBSYSMGMT screen, the user hit PF7/CMDprv and did not specify a name or pattern in the Subsystem Name field.

SUM096 Refine criteria: # of subsystems classes exceeds max(1000).

On the Command Privileges screen, the max number of subsystem classes that may be processed at one time is 1000. If a user specifies a pattern in the Subsystem Name field that results in a list that exceeds 1000, then this message is displayed.

SUM098 Press 12/END to make changes or 3/QUIt to exit.

On the Command Privilege screen, this message is displayed when a user hits PF3/QUIt in order to confirm that user wants to quit without any updates applied.

SUM099 No value specified for Subsystem Name.

On the Command Privilege screen, this message is displayed when the user hits PF4/LISt and does not supply any value in the Subsystem Name field.

SUM099 No value specified for Subsystem Class.

On the Command Privilege screen, this message is displayed when the user hits PF4/LISt and does not supply any value in the Subsystem Class field.

SUM099 SYNTAX ERROR IN PATTERN '<pattern>' NEAR CHARACTER <n>

On the Command Privilege screen, this message is displayed when the user hits PF4/LISt and supplies an invalid pattern the Subsystem Name or Subsystem Class fields.

SUM100 Enter Y or N. Or leave blank if privilege should not be updated.

On the Command Privilege screen, the user hit PF12/END to update the Command Privilege and one of the Command Privileges fields was set to an invalid value.

SUM203 Value tagged is not valid or beyond acceptable range.

On the Command Privilege screen, the user entered an invalid value in the selection field for the Subsystem Name or Subsystem Class. The only valid value for the selection field is “x”.

SUM301 <n> Subsystem(s) enqueued. Press 6/DISplay to list.

On the Command Privilege screen, this message is displayed to inform the user that there are a number of subsystems that are not displayed on the list and cannot be updated do to enqueuing conflicts.

SUM302 Classes enqueued, 6/DISplay or 12/END to perform partial update.

On the Command Privilege screen, this message is displayed to inform the user that there are a number of classes that cannot be updated at this time due to enqueuing conflicts. The user has the option to perform a partial update such that the Command Privileges for the classes that not enqueued are updated. In addition, the user has the option to display the list of classes that are enqueued and can not be updated.

SUM303 <n> Class(es) about to be updated, press 12/END to confirm update

On the Command Privilege screen, this message is displayed when the user presses PF12/END in order to inform the user about the number of Subsystem Classes that are about to be updated.

SUM304 No update, Command Privileges not changed for selected subsystems

On the Command Privilege screen, the user hit PF12/END to update the Command Privileges but either the privileges were not changed or there were no Subsystem Classes that were selected for update.

SUM305 Warning! 11/UPDate must be pressed now to save changes.

On the Command Privilege screen, the user pressed PF14/LIST to obtain a new list but the command privileges for the previous list were not saved.

SUM306 Classes enqueued, 6/DISplay or 11/UPDate to perform partial update.

On the Command Privilege screen, this message is displayed to inform the user that there are a number of classes that cannot be updated at this time due to enqueuing conflicts. The user has the option to perform a partial update such that the Command Privileges for the classes that not enqueued are updated. In addition, the user has the option to display the list of classes that are enqueued and can not be updated.

SUM307 <n> Class(es) about to be updated, press 11/UPDate to confirm update.

On the Command Privilege screen, this message is displayed when the user hits PF6/UPDATE in order to inform the user about the number of Subsystem Classes that are about to be updated.

SUM308 Must use 4/LISt when changing Subsystem or Class patterns

The user entered a new value in the Subsystem Name or Subsystem Class field and pressed an invalid key.

File Management error messages

FIM131 The file is being changed by (userid).

On the main FILEMGMT screen, the user attempted to update a file that is already being changed by (userid).

FIM310 Datetime Stamp parameter (DTSFN) not set in this system.

In order to use the date/time stamp option, the parameter DTSFN must be set in the CCAIN to the fieldname that will be used for date/time stamp processing.

FIM159 Field is Datetime Stamp, field can not be changed.

Date/time stamp field corresponds to value of DTSFN and can not be changed.

FIM160 There is no valid Datetime Stamp field defined in file.

The user is attempting to use a date/time stamp feature, but the file does not have the field defined.

FIM314 You have selected Datetime Stamp, press PF4 to define fields.

On the Reset screen, the user selected the date/time stamp option. This message directs the user to now go define the attributes associated with the date/time stamp field.

FIM315 Warn: Must define field for Datetime Stamp.

The user set FOPT=x'10', indicating a desire to use date/time stamp processing, but a field which corresponds to the value of DTSFN has not been defined to the file.

FIM316 This file is already defined as a Datetime Stamp file.

The user is attempting to define the file as a date/time stamp file, but it already is a date/time stamp file.

FIM319 Enable Datetime Stamp must be a Y or N

The user entered an invalid character in the Enable date/time stamp input area.

FIM320 Can not disable Lock Pending Updates with Datetime Stamp.

The user attempted to select Disable Lock Pending Updates for a date/time stamp file.

FIM469 WARN: A default length of 6000 will be stored for LOB sizing.

On the field attributes screen, when defining a LOB entry, if a length is not added for the field, a default of 6000 will be added.

FIM514 Datetime Stamp field can not be UNIQUE.

On the field attributes screen, when defining a date/time stamp field, the attribute UNIQUE is not allowed.

FIM515 Datetime Stamp field can not be BINARY.

On the field attributes screen, when defining a date/time stamp field, the attribute BINARY is not allowed.

FIM618 Date Time Stamp cannot be deleted from record.

On the record definition screen, when attempting to delete a date/time stamp field from a record, the field can not be deleted.

FIM814 Insufficient FREESIZE for Table E increase.

On the sizing screen, an increase in ESIZE must be less than or equal to what is available in FREESIZE.

FIM941 Blinking fld is AVG LEN MULTIPLIER, H=Hundreds, T=Thousands, M=Millions.

On the field attributes screen, when defining a LOB entry, there is a multiplier that if entered as an H, multiplies the length by 100. If entered as a T, multiplies the length by 1000. Finally if entered as a M, multiplies the length by 1000000.

FIM942 Warn: ESIZE must be greater than 20.

The minimum size for a blob is > 20.

FIM943 Calculated ESIZE to large – maximum assumed.

ESIZE has been set to it’s highest value allowed.

FIM944 Datetime Stamp field can not be INVISIBLE.

On the field attributes screen, when defining a date/time stamp field, the attribute INVISIBLE is not allowed.

FIM945 Datetime Stamp must be STRING.

On the field attributes screen, when defining a date/time stamp field, the attribute STRING is required.

FIM946 Datetime Stamp field can not be multiply occurring.

On the field attributes screen, when defining a date/time stamp field, the attribute OCCURS must be one.

FIM947 Only values of 0 or 1 are valid for RECRDOPT.

On the sizing screen, only values of 0 or 1 are valid for RECRDOPT.

FIM948 XSIZE value is outside the range of allowed values, 0 to 536870911.

On the sizing screen, XSIZE must be between 0 and 536870911.

FIM949 File must be recreated to change XSIZE.

On the sizing screen, when updating sizing information, if you now want to use Table X, the file must be recreated.

FIM950 XRESERVE, XRECPPG and RECRDOPT not valid without XSIZE.

On the sizing screen, XSIZE must have a non-zero value before setting XRESERVE, XRECPPG or RECRDOPT.

FIM951 XRESERVE, XRECPPG must be > 0 when XSIZE is > 0.

If XSIZE is a positive non-zero number, then XRESERVE and XRECPPG must also be positive, non-zero numbers.

FIM952 Insufficient FREESIZE for TABLEX increase.

When attempting to increase the size of Table X, there must be sufficient pages available in FREESIZE to handle the increase.

FIM953 XRESERVE valid values from 0 to 7168.

On the sizing screen, XRESERVE must be between 0 and 7168.

A

Statistics in 64-bit Architecture

In this appendix

- Overview
- User report
- System report
- File statistics
- MP subtask statistics and offsets
- Disk buffer monitor statistics

Overview

The tables in this appendix are provided for your use in identifying and tracking statistics that are reported by Model 204 in various situations. The offsets, given in decimal and hexadecimal, are to help you locate particular statistics within statistical output.

The tables in this appendix reflect work done specifically for Model 204 V7R1.0. The complete collection of tables that identify statistics is found in the Model 204 *System Manager's Guide*.

User report

Table A-1 lists the offset, length, data type, and name of the Model 204 user statistics.

Table A-1. Offset location of SMF logout record

Offset dec(hex)	Length	Data type	Description
60(3C)	4	Binary	CNCT
64(40)	4	Binary	DKRD
68(44)	4	Binary	DKWR
72(48)	4	Binary	SQRD
76(4C)	4	Binary	SQWR
80(50)	4	Binary	SGMTI
84(54)	4	Binary	SGMTO
88(58)	4	Binary	SVRD
92(5C)	4	Binary	SVWR
96(60)	4	Binary	CPU
100(64)	4	Binary	REQ
104(68)	4	Binary	MOVE
108(6C)	4	Binary	DUMP
112(70)	4	Binary	REST
116(74)	4	Binary	SLIC
120(78)	4	Binary	AUDIT
124(7C)	4	Binary	WAIT
128(80)	4	Binary	FBWT
132(84)	4	Binary	UDD
136(88)	4	Binary	RECADD
140(8C)	4	Binary	RECDEL
144(90)	4	Binary	BADD
148(94)	4	Binary	BDEL
152(98)	4	Binary	BCHG
156(9C)	4	Binary	IXADD
160(A0)	4	Binary	IXDEL
164(A4)	4	Binary	FINDS

Table A-1. Offset location of SMF logout record (continued)

Offset dec(hex)	Length	Data type	Description
168(A8)	4	Binary	SORTS
172(AC)	4	Binary	RECDS
176(B0)	4	Binary	STRECDS
180(B4)	4	Binary	DKAR
184(B8)	4	Binary	DKPR
188(BC)	4	Binary	DKRR
192(C0)	4	Binary	COMMITS
196(C4)	4	Binary	BACKOUTS
200(C8)	4	Binary	UPDTTIME(MS)
204(CC)	4	Binary	LONGUPDTS
208(D0)	4	Binary	LONGUPDTIME(MS)
212(D4)	4	Binary	SMPLS
216(D8)	4	Binary	RUNG
220(DC)	4	Binary	REDY
224(E0)	4	Binary	BLKI
228(E4)	4	Binary	WTSV
232(E8)	4	Binary	BLKO
236(EC)	4	Binary	SWPG
240(F0)	4	Binary	PCPU
244(F4)	4	Binary	DIRRCD
248(F8)	4	Binary	BXCHNG
252(FC)	4	Binary	BXDELE
256(100)	4	Binary	BXNEXT
260(104)	4	Binary	BXFIND
264(108)	4	Binary	BXINSE
268(10C)	4	Binary	BXSPLI
272(110)	4	Binary	BXRFND
276(114)	4	Binary	BXFREE
280(118)	4	Binary	STCPU
284(11C)	4	Binary	STDEQ

Table A-1. Offset location of SMF logout record (continued)

Offset dec(hex)	Length	Data type	Description
288(120)	4	Binary	SCHDCPU
292(124)	4	Binary	SCREENS
296(128)	4	Binary	SVPAGES
300(12C)	4	Binary	PBRSFLT
304(130)	4	Binary	MQGETS
308(134)	4	Binary	MQPUTS
312(138)	4	Binary	UBUFHWS
316(13C)	4	Binary	MQHWTASK
320(140)	4	Binary	MQBYTEIN
324(144)	4	Binary	MQBYTEOU
328(148)	4	Binary	MQHWQU
332(14C)	4	Binary	MQNUMQU
336(150)	4	Binary	MQNUMQM
340(154)	4	Binary	MQAPITIM
344(158)	4	Binary	MQAPICNT
348(15C)	4	Binary	MQGWTTIM
352(160)	4	Binary	MQGWTCNT
356(164)	4	Binary	MQGWTTSP
360(168)	4	Binary	MQGWTSUC
364(16C)	4	Binary	ECLOAD
368(170)	4	Binary	ECDELETE
372(174)	4	Binary	ECCALL
376(178)	4	Binary	ECCWAITM
380(17C)	4	Binary	ECCWAITS
384(180)	4	Binary	ECTWAITM
388(184)	4	Binary	ECTWAITS
392(188)	4	Binary	ECCTOUT
396(18C)	4	Binary	ECCNCT
400(190)	4	Binary	GTBLRU
404(194)	4	Binary	GTBLRS

Table A-1. Offset location of SMF logout record (continued)

Offset dec(hex)	Length	Data type	Description
408(198)	4	Binary	FSCBSW
412(19C)	4	Binary	Spare

Table A-2 lists the Since-last record statistics.

Table A-2. SMF since-last record

Offset dec(hex)	Length	Data type	Description
124(7C)	4	Binary	NTBL
128(80)	4	Binary	GTBL
132(84)	4	Binary	QTBL
136(88)	4	Binary	STBL
140(8C)	4	Binary	TTBL
144(90)	4	Binary	VTBL
148(94)	4	Binary	PDL
152(98)	4	Binary	FTBL
156(9C)	4	Binary	XTBL
160(A0)	4	Binary	ITBL
164(A4)	4	Binary	FSCB
168(A8)	4	Binary	OUTPB
172(AC)	4	Binary	HEAP
176(B0)	4	Binary	SQLI
180(B4)	4	Binary	SQLO
184(B8)	4	Binary	CNCT
188(BC)	4	Binary	CPU
192(C0)	4	Binary	DKRD
196(C4)	4	Binary	DKWR
200(C8)	4	Binary	UDD
204(CC)	4	Binary	OUT
208(D0)	4	Binary	SLIC
212(D4)	4	Binary	IN
216(D8)	4	Binary	RECADD

Table A-2. SMF since-last record (continued)

Offset dec(hex)	Length	Data type	Description
220(DC)	4	Binary	RECDEL
224(E0)	4	Binary	BADD
228(E4)	4	Binary	BDEL
232(E8)	4	Binary	BCHG
236(EC)	4	Binary	IXADD
240(F0)	4	Binary	IXDEL
244(F4)	4	Binary	FINDS
248(F8)	4	Binary	SORTS
252(FC)	4	Binary	RECDS
256(100)	4	Binary	STRECDS
260(104)	4	Binary	PCPU
264(108)	4	Binary	RQTM
268(10C)	4	Binary	DIRRCD
272(110)	4	Binary	BXCHNG
276(114)	4	Binary	BXDELE
280(118)	4	Binary	BXNEXT
284(11C)	4	Binary	BXFIND
288(120)	4	Binary	BXINSE
292(124)	4	Binary	BXSPLI
296(128)	4	Binary	BXRFND
300(12C)	4	Binary	BXFREE
304(130)	4	Binary	STCPU
308(134)	4	Binary	STDEQ
312(138)	4	Binary	SCHDCPU
316(13C)	4	Binary	SCREENS
320(140)	4	Binary	SVRD
324(144)	4	Binary	SVWR
328(148)	4	Binary	DKPR
332(14C)	4	Binary	SVPAGES
336(150)	4	Binary	COMMITS

Table A-2. SMF since-last record (continued)

Offset dec(hex)	Length	Data type	Description
340(154)	4	Binary	BACKOUTS
344(158)	4	Binary	UPDTTIME(MS)
348(15C)	4	Binary	LONGUPDTS
352(160)	4	Binary	LONGUPDTIME(MS)
356(164)	4	Binary	MQGETS
360(168)	4	Binary	MQPUTS
364(16C)	4	Binary	UBUFHWS
368(170)	4	Binary	MQHWTASK
372(174)	4	Binary	MQBYTEIN
376(178)	4	Binary	MQBYTEOU
380(17C)	4	Binary	MQHWQU
384(180)	4	Binary	MQNUMQU
388(184)	4	Binary	MQNUMQM
392(188)	4	Binary	MQAPITIM
396(18C)	4	Binary	MQAPICNT
400(190)	4	Binary	MQGWTTIM
404(194)	4	Binary	MQGWTCNT
408(198)	4	Binary	MQGWTTSP
412(19C)	4	Binary	MQGWTSUC
416(1A0)	4	Binary	ECLOAD
420(1A4)	4	Binary	ECDELETE
424(1A8)	4	Binary	ECCALL
428(1AC)	4	Binary	ECCWAITM
432(1B0)	4	Binary	ECCWAITS
436(1B4)	4	Binary	ECTWAITM
440(1B8)	4	Binary	ECTWAITS
444(1BC)	4	Binary	ECCTOUT
448(1C0)	4	Binary	ECCNCT
452(1C4)	4	Binary	GTBLRU
456(1C8)	4	Binary	GTBLRS

Table A-2. SMF since-last record (continued)

Offset dec(hex)	Length	Data type	Description
460(1CC)	4	Binary	FSCBSW
464(1D0)	4	Binary	Spare

Table A-3 list the conflict statistics.

Table A-3. Conflict statistics

Offset dec(hex)	Length	Data type	Description
468(1D4)	4	Binary	BLKCFRE
472(1D8)	4	Binary	BLKRLK
476(1DC)	4	Binary	WTCFR
480(1E0)	4	Binary	WTRLK

Table A-4 lists the user final and partial statistics.

Table A-4. User final and partial statistics

Offset dec(hex)	Length	Data type	Description
40(28)	4	Binary	CNCT
44(2C)	4	Binary	DKRD
48(30)	4	Binary	DKWR
52(34)	4	Binary	SQRD
56(38)	4	Binary	SQWR
60(3C)	4	Binary	SGMTI
64(40)	4	Binary	SGMTO
68(44)	4	Binary	SVRD
72(48)	4	Binary	SVWR
76(4C)	4	Binary	CPU
80(50)	4	Binary	REQ
84(54)	4	Binary	MOVE
88(58)	4	Binary	DUMP
92(5C)	4	Binary	REST
96(60)	4	Binary	SLIC
100(64)	4	Binary	AUDIT
104(68)	4	Binary	WAIT

Table A-4. User final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
108(6C)	4	Binary	FBWT
112(70)	4	Binary	UDD
116(74)	4	Binary	RECADD
120(78)	4	Binary	RECDEL
124(7C)	4	Binary	BADD
128(80)	4	Binary	BDEL
132(84)	4	Binary	BCHG
136(88)	4	Binary	IXADD
140(8C)	4	Binary	IXDEL
144(90)	4	Binary	FINDS
148(94)	4	Binary	SORTS
152(98)	4	Binary	RECDS
156(9C)	4	Binary	STRECDS
160(A0)	4	Binary	DKAR
164(A4)	4	Binary	DKPR
168(A8)	4	Binary	DKRR
172(AC)	4	Binary	COMMITS
176(B0)	4	Binary	BACKOUTS
180(B4)	4	Binary	UPDTTIME(MS)
184(B8)	4	Binary	LONGUPDTS
188(BC)	4	Binary	LONGUPDTIME(MS)
192(C0)	4	Binary	SMPLS
196(C4)	4	Binary	RUNG
200(C8)	4	Binary	REDY
204(CC)	4	Binary	BLKI
208(D0)	4	Binary	WTSV
212(D4)	4	Binary	BLKO
216(D8)	4	Binary	SWPG
220(DC)	4	Binary	PCPU
224(E0)	4	Binary	DIRRCD

Table A-4. User final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
228(E4)	4	Binary	BXCHNG
232(E8)	4	Binary	BXDELE
236(EC)	4	Binary	BXNEXT
240(F0)	4	Binary	BXFIND
244(F4)	4	Binary	BXINSE
248(F8)	4	Binary	BXSPLI
252(FC)	4	Binary	BXRFND
256(100)	4	Binary	BXFREE
260(104)	4	Binary	STCPU
264(108)	4	Binary	STDEQ
268(10C)	4	Binary	SCHDCPU
272(110)	4	Binary	SCREENS
276(114)	4	Binary	SVPAGES
280(118)	4	Binary	PBRSFLT
284(11C)	4	Binary	MQGETS
288(120)	4	Binary	MQPUTS
292(124)	4	Binary	UBUFHWS
296(128)	4	Binary	MQHWTASK
300(12C)	4	Binary	MQBYTEIN
304(130)	4	Binary	MQBYTEOU
308(134)	4	Binary	MQHWQU
312(138)	4	Binary	MQNUMQU
316(13C)	4	Binary	MQNUMQM
320(140)	4	Binary	MQAPITIM
324(144)	4	Binary	MQAPICNT
328(148)	4	Binary	MQGWTTIM
332(14C)	4	Binary	MQGWTCNT
336(150)	4	Binary	MQGWTTSP
340(154)	4	Binary	MQGWTSUC
344(158)	4	Binary	ECLOAD

Table A-4. User final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
348(15C)	4	Binary	ECDELETE
352(160)	4	Binary	ECCALL
356(164)	4	Binary	ECCWAITM
360(168)	4	Binary	ECCWAITS
364(16C)	4	Binary	ECTWAITM
368(170)	4	Binary	ECTWAITS
372(174)	4	Binary	ECCTOUT
376(178)	4	Binary	ECCNCT
380(17C)	4	Binary	GTBLRU
384(180)	4	Binary	GTBLRS
388(184)	4	Binary	FSCBSW
392(188)	4	Binary	Spare

System report

Table A-5 lists system final and partial statistics.

Table A-5. System final and partial statistics

Offset dec(hex)	Length	Data type	Description
16(10)	8	Binary	AUDIT
24(18)	8	Binary	OUT
32(20)	8	Binary	IN
40(28)	8	Binary	OUTXX
48(30)	8	Binary	INXX
56(38)	8	Binary	DEV5
64(40)	8	Binary	DEV6
72(48)	8	Binary	DEV7
80(50)	8	Binary	DEV8
88(58)	8	Binary	DEV9
96(60)	8	Binary	DEV10
104(68)	8	Binary	DEV11
112(70)	8	Binary	DEV12
120(78)	8	Binary	DEV13
128(80)	8	Binary	DEV14
136(88)	8	Binary	OUTTTY
144(90)	8	Binary	INTTY
152(98)	8	Binary	DEV17
160(A0)	8	Binary	DEV18
168(A8)	8	Binary	DEV19
176(B0)	8	Binary	DEV20
184(B8)	8	Binary	DEV21
192(C0)	8	Binary	DEV22
200(C8)	8	Binary	DEV23
208(D0)	8	Binary	DEV24
216(D8)	8	Binary	DEV25
224(E0)	8	Binary	DEV26
232(E8)	8	Binary	DEV27

Table A-5. System final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
240(F0)	8	Binary	DEV28
248(F8)	8	Binary	OUTCRAM
256(100)	8	Binary	INCRAM
264(108)	8	Binary	DEV31
272(110)	8	Binary	DEV32
280(118)	8	Binary	DEV33
288(120)	8	Binary	DEV34
296(128)	8	Binary	DEV35
304(130)	8	Binary	DEV36
312(138)	8	Binary	DEV37
320(140)	8	Binary	DEV38
328(148)	8	Binary	OUTVMIO
336(150)	8	Binary	INVMIO
344(158)	8	Binary	OUTVMFS
352(160)	8	Binary	INVMFS
360(168)	8	Binary	OUTVMIF
368(170)	8	Binary	INVMIF
376(178)	8	Binary	OUTCMIO
384(180)	8	Binary	INCMIO
392(188)	8	Binary	OUTCMFS
400(190)	8	Binary	INCMFS
408(198)	8	Binary	DEV49
416(1A0)	8	Binary	DEV50
424(1A8)	8	Binary	OFFIN
432(1B0)	8	Binary	OFFOU
440(1B8)	8	Binary	DEV53
448(1C0)	8	Binary	DEV54
456(1C8)	8	Binary	DEV55
464(1D0)	8	Binary	DEV56
472(1D8)	8	Binary	DEV57

Table A-5. System final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
480(1E0)	8	Binary	DEV58
488(1E8)	8	Binary	DEV59
496(1F0)	8	Binary	DEV60
504(1F8)	8	Binary	DEV61
512(200)	8	Binary	DEV62
520(208)	8	Binary	DEV63
528(210)	8	Binary	DEV64
536(218)	8	Binary	DEV65
544(220)	8	Binary	DEV66
552(228)	8	Binary	DEV67
560(230)	8	Binary	DEV68
568(238)	8	Binary	DEV69
576(240)	8	Binary	DEV70
584(248)	8	Binary	DEV71
592(250)	8	Binary	DEV72
600(258)	8	Binary	DEV73
608(260)	8	Binary	DEV74
616(268)	4	Binary	WAIT
620(26C)	4	Binary	MPLKWTIM
624(270)	8	Binary	DKRD
632(278)	8	Binary	DKWR
640(280)	4	Binary	SVRD
644(284)	4	Binary	SVWR
648(288)	8	Binary	CPU
656(290)	8	Binary	REQ
664(298)	8	Binary	MOVE
672(2A0)	4	Binary	DUMP
676(2A4)	4	Binary	REST
680(2A8)	4	Binary	SLIC
684(2AC)	4	Binary	CNCT

Table A-5. System final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
688(2B0)	4	Binary	FBWT
692(2B4)	4	Binary	SWT
696(2B8)	4	Binary	ERRPDL
700(2BC)	4	Binary	MPLKPREM
704(2C0)	8	Binary	RECADD
712(2C8)	8	Binary	RECDEL
720(2D0)	8	Binary	BADD
728(2D8)	8	Binary	BDEL
736(2E0)	8	Binary	BCHG
744(2E8)	8	Binary	IXADD
752(2F0)	8	Binary	IXDEL
760(2F8)	8	Binary	FINDS
768(300)	4	Binary	SORTS
772(304)	4	Binary	Spare
776(308)	8	Binary	RECDS
784(310)	8	Binary	STRECDS
792(318)	8	Binary	DKAR
800(320)	8	Binary	DKPR
808(328)	4	Binary	DKRR
812(32C)	4	Binary	FBMX
816(330)	4	Binary	TFMX
820(334)	4	Binary	USMX
824(338)	4	Binary	SVMX
828(33C)	4	Binary	Spare
832(340)	8	Binary	APSYLD
840(348)	8	Binary	APSYLDD
848(350)	4	Binary	APSYLDT
852(354)	4	Binary	Spare
856(358)	8	Binary	DKPRF
864(360)	4	Binary	SMPLS

Table A-5. System final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
868(364)	4	Binary	USRS
872(368)	4	Binary	SVAC
876(36C)	4	Binary	RUNG
880(370)	4	Binary	REDY
884(374)	4	Binary	BLKI
888(378)	4	Binary	WTSV
892(37C)	4	Binary	BLKO
896(380)	4	Binary	SWPG
900(384)	4	Binary	PCPU
904(388)	4	Binary	DIRRCD
908(38C)	4	Binary	Spare
912(390)	8	Binary	STCPU
920(398)	8	Binary	STDEQ
928(3A0)	8	Binary	STWAIT
936(3A8)	4	Binary	STPOST
940(3AC)	4	Binary	LKWAIT
944(3B0)	4	Binary	LKPOST
948(3B4)	4	Binary	RSXCOMP
952(3B8)	4	Binary	SCHDCPU
956(3BC)	4	Binary	SCREENS
960(3C0)	4	Binary	STIMERS
964(3C4)	4	Binary	Spare
968(3C8)	8	Binary	SVPAGES
976(3D0)	4	Binary	PBRSFLT
980(3D4)	4	Binary	COMMITS
984(3D8)	4	Binary	BACKOUTS
988(3DC)	4	Binary	LONGUPDTS
992(3E0)	4	Binary	LONGUPDTIME(MS)
996(3E4)	4	Binary	MPHASHD
1000(3E8)	4	Binary	CDLWAIT

Table A-5. System final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
1004(3EC)	4	Binary	MQGETS
1008(3F0)	4	Binary	MQPUTS
1012(3F4)	4	Binary	UBUFHWS
1016(3F8)	4	Binary	MQHWTASK
1020(3FC)	4	Binary	Spare
1024(400)	8	Binary	MQBYTEIN
1032(408)	8	Binary	MQBYTEOU
1040(410)	4	Binary	MQHWQU
1044(414)	4	Binary	MQNUMQU
1048(418)	4	Binary	MQNUMQM
1052(41C)	4	Binary	Spare
1056(420)	8	Binary	MQAPITIM
1064(428)	8	Binary	MQAPICNT
1072(430)	8	Binary	MQGWTTIM
1080(438)	4	Binary	MQGWTCNT
1084(43C)	4	Binary	Spare
1088(440)	8	Binary	MQGWTTSP
1096(448)	4	Binary	MQGWTSUC
1100(44C)	4	Binary	ECLOAD
1104(450)	4	Binary	ECDELETE
1108(454)	4	Binary	Spare
1112(458)	8	Binary	ECCALL
1120(460)	4	Binary	ECCWAITM
1124(464)	4	Binary	ECCWAITS
1128(468)	4	Binary	ECTWAITM
1132(46C)	4	Binary	ECTWAITS
1136(470)	4	Binary	ECCTOUT
1140(474)	4	Binary	ECCNCT
1144(478)	4	Binary	ECMODMAX
1148(47C)	4	Binary	ECNAMMAX

Table A-5. System final and partial statistics (continued)

Offset dec(hex)	Length	Data type	Description
1152(480)	4	Binary	ECTSKMAX
1156(484)	4	Binary	GTBLRU
1160(488)	4	Binary	GTBLRS
1164(48C)	4	Binary	TSMX
1168(490)	4	Binary	TEMX
1172(494)	4	Binary	MAXIOX
1176(498)	8	Binary	DKRDL
1184(4A0)	8	Binary	DKWRL
1192(4A8)	8	Binary	Spare
1200(4B0)	8	Binary	Spare
1208(4B8)	8	Binary	Spare
1216(4C0)	8	Binary	Spare
1224(4C8)	8	Binary	Spare
1232(4D0)	8	Binary	Spare
1240(4D8)	8	Binary	Spare

Table A-6 lists the system performance statistics.

Table A-6. System performance statistics

Offset dec(hex)	Length	Data type	Description
16(10)	4	Binary	SMPLS
20(14)	4	Binary	USRS
24(18)	4	Binary	SVAC
28(1C)	4	Binary	RUNG
32(20)	4	Binary	REDY
36(24)	4	Binary	BLKI
40(28)	4	Binary	WTSV
44(2C)	4	Binary	BLKO
48(30)	4	Binary	SWPG

File statistics

Table A-7 lists the file statistics

Table A-7. File statistics

File statistic	Offset dec(hex)	Length
DKRD	26(1A)	8
DKWR	34(22)	8
REQ	42(2A)	8
RETRYA	50(32)	8
RETRYC	58(3A)	8
DUPDTS	66(42)	8
RECADD	74(4A)	8
RECDEL	82(52)	8
BADD	90(5A)	8
BDEL	98(62)	8
BCHG	106(6A)	8
IXADD	114(72)	8
IXDEL	122(7A)	8
DIRRCD	130(82)	8
BXCHNG	138(8A)	8
BXDELE	146(92)	8
BXNEXT	154(9A)	8
BXFIND	162(A2)	8
BXINSE	170(AA)	8
BXSPLI	178(B2)	8
BXRFND	186(BA)	8
BXFREE	194(C2)	8
UPDTTIME	202(CA)	8
PNDGTIME	210(D2)	8
DKUPTIME	218(DA)	8
COMMITS	226(E2)	8
BACKOUTS	234(EA)	8

MP subtask statistics and offsets

Table A-8 lists the MP subtask statistics and offsets.

Table A-8. MP subtask statistics and offsets

MP statistic	Offset dec(hex)	Length
CPU	16(10)	8
PR	32(20)	8
PCPU	48(30)	4
SPARE	56(38)	4
STDEQ	64(40)	8
STWAIT	80(50)	4
STPOST	88(58)	4
LKWAIT	96(600)	4
LKPOST	104(68)	4
MQWTM	112(70)	8

Disk buffer monitor statistics

Table A-9 lists the disk buffer monitor statistics and their offsets.

Table A-9. Disk buffer monitor statistics

Offset dec(hex)	Length	Data type	Disk buffer monitor statistic
16(10)	8	Binary	DKSWAIT
24(18)	8	Binary	DKSTKQC
32(20)	8	Binary	DKSWRP
40(28)	8	Binary	DKSWRPT
48(30)	8	Binary	DKSDIR
56(38)	8	Binary	DKSDIRT
64(40)	8	Binary	DKSKIP
72(48)	8	Binary	DKSRHC
80(50)	8	Binary	DKSFNU
88(58)	8	Binary	DKSRR
96(60)	8	Binary	DKSFBS
104(68)	8	Binary	DKSKIPT
112(70)	8	Binary	DKSAWT
120(78)	8	Binary	DKSAWW
128(80)	8	Binary	DKSWWL
136(88)	8	Binary	DKSAWB
144(90)	8	Binary	DKSAWBL
152(98)	8	Binary	DKSRRFND
160(A0)	8	Binary	DKSTBLF
168(A8)	8	Binary	DKSTBLA
176(B0) 8	8	Binary	DKSTBLB
184(B8)	8	Binary	DKSTBLC
192(C0)	8	Binary	DKSTBLD
200(C8)	8	Binary	Spare
208(D0)	8	Binary	Spare
216(D8)	8	Binary	Spare
224(E0)	8	Binary	Spare
232(E8)	8	Binary	Spare

Table A-9. Disk buffer monitor statistics (continued)

Offset dec(hex)	Length	Data type	Disk buffer monitor statistic
240(F0)	8	Binary	Spare
248(F8)	8	Binary	Spare
256(100)	8	Binary	DKSTBLX
264(108)	8	Binary	DKSTBLE
272(110)	8	Binary	Spare

Index

Symbols

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z