Rocket M204 SirAud

Installation and Reference Guide

Version 1.5

September 2013
AUD-0105-IR-01

Notices

Edition

Publication date: September 2013

Book number: AUD-0105-IR-01

Product version: Rocket M204 SirAud Version 1.5

Copyright

Trademarks

Examples

This information might contain examples of data and reports. The examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

License agreement

Contact information

Contacting Global Technical Support

If you have current support and maintenance agreements with Rocket Software and CCA, contact Global Technical Support by email or by telephone:

Email: m204support@rocketsoftware.com

Telephone:

North America	+1 800 755 4222
United Kingdom/Europe	+44 (0) 20 8867 6153

Alternatively, you can access the Rocket Customer Portal and report a problem, download an update, or read answers to FAQs. You will be prompted to log in with the credentials supplied as part of your product maintenance agreement.

To log in to the Rocket Customer Portal, go to:

www.rocketsoftware.com/support

Contents

Contents

Summary of Changes

This section describes significant changes to the documentation. In most cases these changes correspond to enhancements made to the underlying product.

SirAud Version 1.04

The 1.04 version is a maintenance release only.

SirAud Version 1.03

No version 1.03 was released.

SirAud Version 1.02

- New FROM and TO parameters. FILERP, PERFRP, PROCRP, SUBRP, UTBLRP, and UTBLSM parameters, for controlling which reports to produce, and producing all output on the SYSPRINT file and only there.

- Zaps for customization of defaults

SirAud Version 1.01

Base version of this manual.

CHAPTER 1 *Overview*

SirAud reads a single journal file produced by *Model 204*, and it generates a set of reports, using statistics entries as well as other information in the journal. These reports are intended to provide an overview of the resource usage for one *Model 204* run. Each report summarizes a different aspect of the run.

1.1 Versions and Compatibility

Any configuration that runs correctly with a version of *SirAud* will run correctly on subsequent versions of *SirAud*. This document, the *SirAud Installation and Reference Guide*, assumes that a site is running *SirAud* version 1.01 or later. Any documentation that requires a later version of *SirAud* will be clearly marked to indicate this. If a feature is not indicated as requiring any specific version of *SirAud*, it can be assumed that it is available as documented in all versions of *SirAud*; that is, all versions since version 1.01.

CHAPTER 2 *SirAud Reports*

A report is generated when a corresponding file is defined or when the report is requested by a *SirAud* program parameter. The sections in this chapter give the report names and a brief description of the report contents.

If a report is selected by specifying a *SirAud* program parameter (see "ALLPR and NOPR" on page 7), all reports are produced on the SYSPRINT file, and only there. Otherwise, the name of the report given in the section headings in this chapter is the same as the name of the DD statement (in MVS, or the FILEDEF command, in CMS) to which the report is written.

2.1 PERFREPT

This report summarizes system-level performance over each ACCTIM interval.

2.2 FILEREPT

This report summarizes the resource utilization of each *Model 204* file. Files are reported in descending order by total disk I/O (DKRD+DKWR).

2.3 PROCREPT

This report summarizes the resource utilization of each *Model 204* procedure. Procedures are reported by subsystem and file, in descending order by total CPU.

2.4 SUBREPT

This report summarizes the resource utilization of each *Model 204* subsystem. Subsystems are reported in descending order by total CPU.

2.5   UTBLREPT

This report summarizes the current high water mark of each user-resettable server table on a subsystem basis. A set of "target" table settings is also presented. Subsystems are reported in descending order by the largest server size needed to run all of the subsystem procedures executed during the run. A separate line is produced for each procedure in the subsystem that has a table high water mark that is above the "target" value.

2.6   UTBLSUMM

This report summarizes the current high water mark of each user-resettable server table for each subsystem. Subsystems are reported in descending order by the largest server size needed to run all of its procedures executed during the run.

Additional SirAud Files

3.1   JOURNAL

This is the input *Model 204* journal file.

3.2   SYSPRINT

Various information is written to this file; the following information is always written to SYSPRINT:

- Various **AUD**nnn messages, which are explained in "SirAud Messages" on page 27.

- Identification of the *Model 204* run that created the JOURNAL file.

- A list of pertinent run-time parameters as specified in the *Model 204* run's EXEC, user 0, and user n parameters.

Also, if any *SirAud* reports are specified as program parameters, reports are contained in the SYSPRINT file, and only there; see "ALLPR and NOPR" on page 7.

3.3   SYSIN

Prior to version 1.05 of *SirAud*, no SYSIN file contents were read by *SirAud*. As of version 1.05, all *SirAud* program parameters ("SirAud Parameters" on page 7) may be specified on input records in the SYSIN dataset, which will be read and parsed under MVS and CMS.

You can specify multiple parameters on one SYSIN record, and you can specify multiple records, but you cannot specify a parameter on one record and "continue" its value(s) on another record.

The following simple example of SYSIN content shows valid SIRAUD parameter specifications:

```
FROM Ø4169 1Ø:ØØ
PERFRP
TO Ø417Ø 12:ØØ
```

SirAud Parameters

SirAud has a set of parameters you can use to restrict the size of some reports or otherwise configure the *SirAud* run. These parameters must be provided in either of these ways:

- As a PARM on the EXEC card in MVS, for example:

      //SIRAUD   EXEC PGM=SIRAUD,PARM='NSUBS=1Ø ALLPR'

- As options on the M204CMS command in CMS, for example:

      'M2Ø4CMS SIRAUD (NSUBS 1Ø ALLPR %'

Under MVS, parameters are specified either by a parameter name alone or by a parameter name followed by an equal sign (=) followed by the parameter value. Under CMS, parameters are specified by a parameter name followed by a space and a percent sign (%), or by a parameter name followed by a space and a parameter value.

This chapter describes the parameter settings that are available for *SirAud*.

Note: You can change the default setting for some of the parameters; see "Customization of Defaults" on page 35.

4.1 ALLPR and NOPR

This parameter is one of the report selection parameters. The report selection parameters determine which reports are produced. They also direct all of the report output of *SirAud* to be written to the SYSPRINT output file, rather than having each of *SirAud*'s reports written to a separate output file. If any report selection parameter is specified, all of the *SirAud* report output is written to SYSPRINT, and only to there.

The report selection parameters are ALLPR, NOPR, and the following:

      FILERP and NOFILERP
      PERFRP and NOPERFRP
      PROCRP and NOPROCRP
      SUBRP and NOSUBRP
      UTBLRP and NOUTBLRP
      UTBLSM and NOUTBLSM

Any of the above individual report parameters beginning with "NO" is called the "negative" form of the parameter, and it indicates that the given report should not be produced The form without "NO" is called the "positive" form, and it indicates the given report should be produced.

ALLPR is a shorthand for a list of all of the positive forms of the individual report parameters. NOPR is a shorthand for a list of all of the negative forms of the individual report parameters.

The effect of the report selection parameters is cumulative, and the specification of any of them can override the effect of a preceding specification. If the first report selection parameter specified is a positive individual report parameter, an implicit NOPR precedes that. If the first report selection parameter specified is a negative individual report parameter, an implicit ALLPR precedes that.

Thus, for example, the following produces all reports except PERFREPT and FILEREPT:

```
PARM=(NOPERFRP,NOFILERP)
```

The following produces only the UTBLSUMM and SUBREPT reports:

```
PARM=(UTBLSM,SUBRP)
```

And the following produces all reports except PROCREPT (the UTBLR in this example is superfluous):

```
PARM=(NOPROCRP,UTBLRP)
```

You may customize your *SirAud* load module so that ALLPR is the default; see "ALLPR parameter customization" on page 35.

The ALLPR and NOPR parameters are new in version 1.02, as are all the report selection parameters.

4.2   FROM=yyddd | time | yyddd time

New in version 1.02, this parameter is used to restrict the analysis of the journal to those journal blocks that contain records that are at or after the indicated date and time. If the time on the journal block's trailer is greater than or equal to the FROM specification, the records in that block are included in the report analysis. Consequently, some records that are earlier than the FROM setting may be included.

The FROM parameter options are:

yyddd   Indicates the date, in Julian format, where the FROM century is such that the start of the time period is within 40 years of the start date of the journal.

The analysis starts at the first journal record in the first journal block on or after the indicated day.

time   Indicates the 24-hour time in the format hh:mm[:ss]; *ss* defaults to 00.

The analysis starts at the first journal record in the first journal block on or after the indicated time of the first day of the journal. If the `time` value is less than the time of the first journal block, the analysis starts at the first journal record in the first journal block on or after the indicated time of the day after the first day of the journal.

yyddd time   Indicates the date, in Julian format, where the FROM century is such that the start of the time period is within 40 years of the start date of the journal, and the 24-hour time in the format hh:mm[:ss] (where *ss* defaults to 00).

The analysis starts at the first journal record in the first journal block on or after the indicated time of the indicated day.

Note: This format cannot be passed as a command line program parameter under CMS, although it may be used in a SYSIN file record (see "SYSIN" on page 5). Under MVS, the format is:

```
..,PARM=' .. FROM=''yyddd time'' .. '
```

Also see the TO parameter ("TO" on page 12), which can be used to specify the end of the date/time range for analysis.

Regardless of any date range restriction specified by FROM or TO, *SirAud* examines the beginning part of the journal, through the message "M204.0075: INITIALIZATION COMPLETED. BUFFERS = n". The histogram of journal record types will include totals from this initialization section of the journal as well as from the date range selected for analysis.

A date range can be useful to provide a reference for interpreting other measures of *Model 204* activity. For example, *SirTune* can show ratios for running proc A versus other procs. Assume that a *SirTune* report over some busy time period shows proc A using 11% of the CPU in that period and then you change the proc. After the changes, *SirTune* on a new busy period might show proc A CPU goes up to 12%. One way to check the effectiveness of your changes is to use *SirAud* PROCREPTs with date/time ranges to see if this increase in CPU correlates to an increased number of invocations between the two busy times.

4.3   FILERP and NOFILERP

The FILERP and NOFILERP parameters have two effects:

1.   Forcing all report output to the SYSPRINT file
2.   Producing (FILERP) or suppressing (NOFILERP) the FILEREPT report.

See "ALLPR and NOPR" on page 7 for a discussion of these and the other report selection parameters.

The FILERP and NOFILERP parameters are new in version 1.02.

4.4   MAXBLK=n

This parameter is used to specify the number of journal blocks to process.

4.5   NAMEOFF=n

This can be used, under the direction of Technical Support, to facilitate running *SirAud* with newer releases of *Model 204*, compensating for changes in the journal format.

4.6   NFILES=n

This parameter is used to restrict the number of files summarized in FILEREPT to the top n files by total disk I/O.

4.7   NPROCS=n

This parameter is used to restrict the number of procedures summarized in PROCREPT to the top n procedures by total CPU.

4.8   NSUBS=n

This parameter is used to restrict the number of subsystems summarized in SUBREPT to the top n subsystems by total CPU.

4.9   NUTBLS=n

This parameter is used to restrict the number of subsystems summarized in UTBLREPT to the top n subsystems by server size.

4.10   NUSUMM=n

This parameter is used to restrict the number of subsystems summarized in UTBLSUMM to the top n subsystems by server size.

4.11   PERFRP and NOPERFRP

The PERFRP and NOPERFRP parameters have two effects:

1.   Forcing all report output to the SYSPRINT file
2.   Producing (PERFRP) or suppressing (NOPERFRP) the PERFREPT report.

See "ALLPR and NOPR" on page 7 for a discussion of these and the other report selection parameters.

The PERFRP and NOPERFRP parameters are new in version 1.02.

4.12   PROCRP and NOPROCRP

The PROCRP and NOPROCRP parameters have two effects:

1.   Forcing all report output to the SYSPRINT file
2.   Producing (PROCRP) or suppressing (NOPROCRP) the PROCREPT report.

See "ALLPR and NOPR" on page 7 for a discussion of these and the other report selection parameters.

The PROCRP and NOPROCRP parameters are new in version 1.02.

4.13   SKIPBLK=n

This parameter is used to specify a number of journal blocks to skip.

4.14   SUBRP and NOSUBRP

The SUBRP and NOSUBRP parameters have two effects:

1. Forcing all report output to the SYSPRINT file
2. Producing (SUBRP) or suppressing (NOSUBRP) the SUBREPT report.

See "ALLPR and NOPR" on page 7 for a discussion of these and the other report selection parameters.

The SUBRP and NOSUBRP parameters are new in version 1.02.

4.15   TO=yyddd | time | yyddd time

New in version 1.02, this parameter is used to restrict the analysis of the journal to those journal blocks that contain records that are at the indicated date and time. If the time on the journal block's header is less than or equal to the TO specification, the records in that block are included in the report analysis. Consequently, some records that are later than the TO setting may be included.

The TO parameter options are:

yyddd   Indicates the date, in Julian format, where the TO century is such that the end of the time period is within 40 years of the start date of the journal.

The analysis ends with the last journal record in the last journal block on or before the indicated day.

time   Indicates the 24-hour time in the format hh:mm[:ss]; *ss* defaults to 59.

The analysis ends with the last journal record in the last journal block on or before the indicated time of the starting day of analysis. If the `time` value is less than the starting time of analysis, the analysis ends with the last journal record in the last journal block on or before the indicated time of the day after the starting day of analysis.

The starting date and time of analysis is specified by the FROM parameter, if present, and otherwise is the date and time at the start of the journal.

yyddd time   Indicates the date, in Julian format, where the TO century is such that the end of the time period is within 40 years of the start date of the journal, and the 24-hour time in the format hh:mm[:ss]; *ss* defaults to 59.

The analysis ends with the last journal record in the last journal block on or before the indicated time of the indicated day.

Note: This format cannot be passed as a command line program parameter under CMS, although it may be used in a SYSIN file record (see "SYSIN" on page 5). Under MVS, the format is:

```
..,PARM=' .. TO=''yyddd time'' .. '
```

Also see the FROM parameter ("FROM" on page 8), which can be used to specify the start of the date/time range for analysis. The description there also provides an example of one benefit of date/time range selection with *SirAud*.

4.16   UTBLRANG=n

This parameter is used to specify the number of standard deviations a server table high water mark for an individual procedure is allowed to vary from the mean for the subsystem before the procedure is flagged for UTBLREPT. The default is 2.

4.17   UTBLRP and NOUTBLRP

The UTBLRP and NOUTBLRP parameters have two effects:

1. Forcing all report output to the SYSPRINT file
2. Producing (UTBLRP) or suppressing (NOUTBLRP) the UTBLREPT report.

See "ALLPR and NOPR" on page 7 for a discussion of these and the other report selection parameters.

The UTBLRP and NOUTBLRP parameters are new in version 1.02.

4.18   UTBLSM and NOUTBLSM

The UTBLSM and NOUTBLSM parameters have two effects:

These parameters have two effects:

1. Forcing all report output to the SYSPRINT file

2. Producing (UTBLSM) or suppressing (NOUTBLSM) the UTBLSUMM report.

See "ALLPR and NOPR" on page 7 for a discussion of these and the other report selection parameters.

The UTBLSM and NOUTBLSM parameters are new in version 1.02.

CHAPTER 5 *Running SirAud under MVS*

The following JCL will run *SirAud* under MVS:

```
//SIRAUDJ  JOB (Ø),MSGCLASS=A,CLASS=A,NOTIFY=ME
//SIRAUD   EXEC PGM=SIRAUD,PARM='NFILES=1Ø'
//STEPLIB  DD DSN=SIRAUD.LOAD,DISP=SHR
//JOURNAL  DD DSN=M2Ø4.CCAJRNL,DISP=SHR
//SYSPRINT DD SYSOUT=A
//* Omit following if reports selected by PARM:
//PERFREPT DD SYSOUT=A
//FILEREPT DD SYSOUT=A
//PROCREPT DD SYSOUT=A
//SUBREPT  DD SYSOUT=A
//UTBLREPT DD SYSOUT=A
//UTBLSUMM DD SYSOUT=A
//
```

Note: If you use any of the report selection parameters, you should omit the various report DD statements, excepting SYSPRINT. See "ALLPR and NOPR" on page 7 for a discussion of the report selection parameters.

CHAPTER 6   *Running SirAud under CMS*

A simple SIRAUD EXEC is provided to run *SirAud*. You must first issue a FILEDEF command for the JOURNAL file. You may want to insert a JOURNAL FILEDEF to identify the *Model 204* journal file, or you may want to issue EXEC SIRAUD from a "wrapper" that contains the FILEDEF command for JOURNAL.

SirAud parameters are specified as options to SIRAUD EXEC or as SYSIN file records (see). Unlike the MVS JCL PARM= field, you do not specify an equal sign (=) between a parameter and the value you are specifying. For example:

```
SIRAUD (NFILES 10 NOSUBREPT
```

If you do not want to use the SIRAUD EXEC that is distributed with the product, *SirAud* must be invoked by the *Model 204* CMS interface, because it uses the CMS interface's QSAM/BSAM simulation. The program name of *SirAud* as distributed is `SIRAUD`.

Note: Because *SirAud* uses the *Model 204* CMS interface, any file can be on an OS format minidisk.

_____
CHAPTER 7 *Report Contents*

7.1 PERFREPT

Each line in a PERFREPT report corresponds to a partial interval as defined by the ACCTIM parameter. The columns in the report are derived from the information in the system partial statistics entries, since last statistics entries for CMPL, LOAD, and EVAL activities, and user logout statistics entries. Unless otherwise noted, statistics values come from the system partial statistics entry.

DATE is the date when the statistics for the interval were logged to the journal.

END OF INTVL
is the time when the statistics for the interval were logged to the journal.

AVG USERS
is defined as the connect time (CNCT) used by all logged in users during the interval, divided by the seconds in the interval. The connect time for a user is reported in the user partial and logout statistics entries.

%CPU is the CPU value divided by the CNCT value.

PCPU is the PCPU value reported for the interval when *Model 204* is running on a single CPU. When MP is active and offload subtask partial statistics are produced, the PCPU for the interval is the CPU for the interval divided by the PR for the interval.

SCREENS/SEC, DKRD+DKWR/SEC, SVRD+SVWR/SEC, LKWAIT+LKPOST/SEC
are the statistics or sum of statistics, divided by the number of seconds in the interval.

JRNL I/O/SEC
is the number of blocks written during the interval, divided by the number of seconds in the interval.

RESP TIME SECS/SCRN MEAN
refers to the mean response time in seconds per screen over the interval. Sample points for determining the mean are gathered from the user since last statistics entries. Each execution of a procedure is accompanied by a CMPL/EVAL or LOAD/EVAL since last pair.

The RQTM statistic reported in the since last entries is summed for a user until an EVAL since last entry is found that has a SCREENS value not equal 0. A sample point is then determined by taking the RQTM sum and dividing

it by SCREENS. In the case when a partial interval completes and a sample point is in progress (no EVAL with SCREENS not equal 0 found yet), the sample point is determined using the RQTM sum through the last EVAL and dividing it by 1. The RQTM from the last CMPL or LOAD since last entry is attributed to the next partial interval.

RESP TIME SECS/SCRN STDEV

refers to the standard deviation of the response time in seconds per screen over the interval. Sample points for determining the standard deviation are the same as with the mean.

CPU SECS/SCRN MEAN

refers to the mean CPU time used in seconds per screen. Sample points for determining the mean are calculated the same way as for the mean response time, except that the CPU statistic is used instead of RQTM.

CPU SECS/SCRN STDEV

refers to the standard deviation of the CPU time used in seconds per screen over the interval. Sample points for determining the standard deviation are the same as with the mean.

DKRD+DKWR/SCRN, SVRD+SVWR/SCRN

are those statistics values divided by the SCREENS for the interval.

7.2   FILEREPT

This report is a subset of statistics from the file close statistics entries and other file information reported in the journal, totalled for each file and for all files. The *Model 204* files are listed in descending order by the sum of the DKRD and DKWR columns.

FILE NAME

is the name of the *Model 204* file. The first line of the report presents the totals for all *Model 204* files and uses *TOTALS* as the name.

DKRD, DKWR, DIRRCD, BXFIND, BXNEXT, BXRFND, BADD+BDEL+BCHG, IXADD+IXDEL, RECADD+RECDEL, BXINSE+BXDELE+BXCHNG, RETRYC

are the totals of the specified statistic or sum of statistics.

DKUPDT   is the number of times pages from the file were flushed out of the buffer pool.

7.3 PROCREPT

One logical line is produced for each unique combination of procedure name, procedure file name, and subsystem name executed during the *Model 204* run, and a summation line is produced for all procedures executed. Each logical line requires two physical lines.

The columns are derived from user since last statistics entries for CMPL, LOAD, or EVAL activities. The procedures are sorted in descending order by TOTAL CPU SECONDS.

RANK   is the rank of the procedure by TOTAL CPU SECONDS.

PROCEDURE NAME
is the name of the procedure.

PROC FILE
is the name of the file to which the procedure belongs.

SUBSYSTEM NAME
is the name of the subsystem to which the procedure file belongs.

NUMBER OF RUNS
is the number of times a procedure was executed.

TOTAL CPU SECS
is the total CPU from the CMPL, LOAD, and EVAL since last entries converted from milliseconds to seconds.

EVAL CPU SECS
is the total CPU from EVAL since last entries converted from milliseconds to seconds.

STCPU/CPU
is the sum of all STCPU divided by the sum of all CPU.

RESP TIME MEAN
refers to the mean response time in seconds per procedure run. An execution of a procedure is accompanied by a CMPL/EVAL or LOAD/EVAL since last pair, and it determines a sample point for calculating the mean response time.

SirAud assigns a variable known as RUNIN to each run. RUNIN is set to the SCREENS from the EVAL since last entry, if SCREENS is not equal 0. Otherwise, RUNIN is set to the IN from the EVAL since last entry, if IN is not equal 0. Otherwise, RUNIN is set to 1.

The RQTM statistic from the CMPL or LOAD since last entry is added to the RQTM from the EVAL. A sample point is then determined by taking the RQTM sum and dividing it by RUNIN.

RESP TIME STDEV

refers to the standard deviation of the response time in seconds per procedure run. Sample points for determining the standard deviation are the same as with the mean.

TOTAL DKRD+DKWR, TOTAL BXFIND, TOTAL BXNEXT

are the totals of the specified statistic or sum of statistics.

UPDATE UNIT SECS MEAN

refers to the mean time in seconds an update unit was active for the procedure. A sample point is the time between the start and end of an update unit.

UPDATE UNIT SECS STDEV

refers to the standard deviation of the number of seconds seconds an update unit was active for the procedure. Sample points for determining the standard deviation are the same as with the mean.

NUMBER OF SCREENS

is the total SCREENS for the totals line. The sum of the RUNIN variable for a procedure is used for an individual procedure.

AVERAGE CPU SECS

is the total CPU divided by NUMBER OF SCREENS, then converted from milliseconds to seconds. All other "AVERAGE" columns are the total of the specified statistic divided by NUMBER OF SCREENS.

AVERAGE DKRD+DKWR, AVERAGE DKPR, AVERAGE SVRD+SVWR, AVERAGE STDEQ, AVERAGE FINDS, AVERAGE DIRRCD, AVERAGE RECDS, AVERAGE SORTS, AVERAGE STRECDS

are the totals of the specified statistic or sum of statistics divided by NUMBER OF SCREENS.

7.4   SUBREPT

There is one line produced for each subsystem that was active during the *Model 204* run and a summation line for all subsystems that were active. The columns are derived from user since last statistics entries for CMPL, LOAD, or EVAL activities for member procedures. The subsystems are sorted in descending order by TOTAL CPU SECONDS.

SUBSYSTEM NAME

is the name of the subsystem. The first line of the report presents the totals for all subsystems and uses *TOTALS* as the name. *NONE* is used for reporting on procedures that were run outside of subsystem context.

TOTAL CPU SECS

is the total CPU from the CMPL, LOAD, and EVAL since last entries converted from milliseconds to seconds.

EVAL CPU SECS

is the total CPU from EVAL since last entries converted from milliseconds to seconds.

STCPU/CPU

is the sum of all STCPU divided by the sum of all CPU.

TOTAL DKRD+DKWR, TOTAL SVRD+SVWR

are the total of the specified statistics sums.

NUMBER OF SCREENS

is the total SCREENS from EVAL since last entries.

NUMBER OF RUNS

is the number of times member procedures were executed.

RESP TIME MEAN

refers to the mean response time in seconds per member procedure run. An execution of a member procedure is accompanied by a CMPL/EVAL or LOAD/EVAL since last pair, and it determines a sample point for calculating the mean response time.

SirAud assigns a variable known as RUNIN to each run. RUNIN is set to the SCREENS from the EVAL since last entry, if SCREENS is not equal 0. Otherwise, RUNIN is set to the IN from the EVAL since last entry, if IN is not equal 0. Otherwise, RUNIN is set to 1.

The RQTM statistic from the CMPL or LOAD since last entry is added to the RQTM from the EVAL. A sample point is then determined by taking the RQTM sum and dividing it by RUNIN.

RESP TIME STDEV

refers to the standard deviation of the response time in seconds per member procedure run. Sample points for determining the standard deviation are the same as with the mean.

CPU SECS/SCRN

is CPU converted to seconds from milliseconds and divided by SCREENS. All other "/SCRN" columns are the statistic value divided by SCREENS.

DKRD+DKWR/SCRN, SVRD+SVWR/SCRN
> are those sum of statistics divided by the SCREENS.

SERVER SIZE (K)
> is the smallest server size, in units of 1024 bytes, needed to execute the member procedures without any UTABLE commands. A server is made up of a fixed portion, determined by *Model 204* user initialization parameters (for example, LOBUFF) that are fixed during the run, and a variable portion containing tables that can be resized using a UTABLE command. The size calculated for this column is based on the member procedures that were actually executed during the run. See "UTBLREPT" for a more detailed explanation.

7.5   UTBLREPT

This report provides a guide to how large a server is actually needed to run a site's application subsystems. There is one set of lines produced for each subsystem that was active during the *Model 204* run. There is a column for each server table that can be reset with a UTABLE command. The values are derived from since last statistics entries for CMPL, LOAD, or EVAL activities for each procedure that is a member of the subsystem. The value reported is the high water mark over all the entries.

SirAud calculates one fixed server size using table sizes specified on the *Model 204* user zero parameter line, the IODEV parameter lines, or the *Model 204* system default. The largest value encountered is used in the calculation. This is different from the way *Model 204* operates in that *SirAud* uses one fixed server size for reporting purposes while *Model 204* calculates a fixed server size for each user. The variable server size is calculated for each subsystem using the aforementioned server table high water marks from the since last statistics entries.

SERVER SIZE (K) column is the sum of the fixed and variable server sizes in units of 1024 bytes. This value shows approximately the largest server needed to execute all the subsystem's procedures that were executed during the *Model 204* run. The subsystems are sorted in descending order by SERVER SIZE (K) for the line identified by *CURRENT*.

The columns other than SERVER SIZE (K) are the high water marks of the corresponding server table as reported in the since last entries for the member procedures that were executed. These columns are: **LFTBL, LGTBL, LITBL, LNTBL, LQTBL, LSTBL, LTTBL, LVTBL, LXTBL, LFSCB**, and **LPDLST**.

SUBSYSTEM is the name of the subsystem. The first line of the report presents the totals for all subsystems and uses *TOTALS* as

the name.   *NONE* is used for reporting on procedures that were run outside of subsystem context.

PROCEDURE NAME   is the name of the member procedure.

PROC FILE   is the name of the file to which the member procedure belongs.

The *SirAud* parameter UTBLRANG is used to identify member procedures that possibly cause the subsystem to require a larger server. For each server table, *SirAud* calculates the mean high water and standard deviation around the mean over all the member procedures that were executed. UTBLRANG specifies the number of standard deviations an individual procedure server table high water mark can stray from the subsystem mean before it is flagged out of range.

Once all server tables of all member procedures have been examined, *SirAud* computes a new set of high water marks and server size using just those procedures whose server table high water marks are in range. These values are reported on the line identified by *TARGET*. The "problem procedures" are then reported individually, sorted by the server size required to execute the procedure and its server table high water marks.

7.6   UTBLSUMM

This report presents the server table high water marks that are used to produce the *CURRENT* lines of UTBLREPT.

Report Contents

APPENDIX A *SirAud Messages*

AUD000   **JOURNAL FILE IS NOT PRESENT**

SirAud was not able to find the JOURNAL file.

AUD001   **UNABLE TO ALLOCATE *n* BYTES ABOVE THE LINE**

SirAud was not able to allocate storage for its internal control blocks in an XA environment.

AUD001   **UNABLE TO ALLOCATE *n* BYTES BELOW THE LINE**

SirAud was not able to allocate storage for its internal control blocks in a non-XA environment.

AUD002   **ERROR PARSING EXECUTE PARAMETERS**

SirAud encountered an error parsing the EXEC parameters echoed in the JOURNAL file.

AUD003   **ERROR PARSING USER PARAMETERS**

SirAud encountered an error parsing the user parameters specified on the user 0 input or user n IODEV line echoed in the JOURNAL file.

AUD004   **UNREADABLE JOURNAL**

SirAud was not able to read the first block of the JOURNAL file.

AUD005   **JOURNAL VERSION MUST BE AT LEAST 2.2.0**

SirAud cannot process a JOURNAL file created by a version of Model 204 before Version 2.2.0.

AUD006   **END OF FILE DETECTED ON JOURNAL**

AUD007   **JOURNAL BLOCK SIZE > DCB BLKSIZE**

SirAud read a JOURNAL block with a size that is greater than the JOURNAL DCB blksize.

AUD008   **FIRST ENTRY NOT A HEADER ENTRY**

This error indicates a serious problem with the JOURNAL file.

AUD009   **DATE/TIME STAMP OUT OF SEQUENCE**

SirAud encountered a date/time stamp sequence error with a JOURNAL block. This error indicates a serious problem with the JOURNAL file.

AUD010   **SEQUENCE COUNTER OUT OF SEQUENCE**

SirAud encountered a block sequence error with a JOURNAL block. This error indicates a serious problem with the JOURNAL file.

AUD011   **RECORD LENGTH BAD OR NO TRAILER ENTRY**

SirAud encountered a format error with a JOURNAL block. This error indicates a serious problem with the JOURNAL file.

AUD012   **PREVIOUS JOURNAL RECORD HEADER**

SirAud dumps the header of the previous JOURNAL block in hexadecimal format for debugging purposes when a sequence error is detected.

AUD013   **CURRENT JOURNAL RECORD HEADER**

SirAud dumps the header of the JOURNAL block in error in hexadecimal format for debugging purposes when a sequence error is detected.

AUD014   **HEX/CHAR DUMP OF RECORD (FIRST 32 BYTES)**

SirAud dumps the first 32 bytes of the JOURNAL block in error in hexadecimal and character format for debugging purposes when a format error is detected.

AUD015   **JOURNAL DOES NOT CONTAIN BEGINNING OF RUN**

This message indicates that the JOURNAL input does not contain the information stored at the beginning of a *Model 204* job. This probably indicates that processing started at a point beyond the actual beginning of the journal.

This message will be followd by message AUD016.

AUD016   **UNABLE TO DECODE CONFIGURATION OF RUN**

This message indicates that *SirAud* did not have the information needed to determine the various *Model 204* initialization settings. *SirAud* will use its own best attempts to guess at settings to use to analyze the journal.

Processing will continue, and the *SirAud* return code will be set to 8 or higher.

AUD019   **MAXBLK PARAMETER EXCEEDED, PROCESSING TERMINATED**

This message indicates that *SirAud* processing is terminated after processing the number of JOURNAL blocks specified in the MAXBLK parameter.

AUD020   **SIRAUD EXPIRED.. *date*.**

This message appears when trying to run *SirAud* after a trial period has expired. The message shows the date of expiration of the trial. Call Technical Support for a new expiration ZAP if the trial has been extended or *SirAud* has been purchased.

AUD021   **SIRAUD NOT AUTHORIZED FOR CPU *cpu_id*.**

This message appears when trying to run *SirAud* on a CPU for which it is not authorized. Call Technical Support for a new CPU ZAP if the new CPU is a replacement for an old one. Note that processor upgrade fees might be required for a new CPU. The *cpu_id* indicated in this message is the processor ID that should be given to Technical Support for a new CPU ZAP.

AUD022   **CALL SIRIUS SOFTWARE FOR A NEW AUTHORIZATION**

This indicates a problem with authorization of *SirAud* on your CPU. Contact Technical Support.

AUD023   **AUTHORIZATION ZAP PRODUCED *date time***

This informational message is issued at the start of a *SirAud* run. It shows the date and time at which the *SirAud* authorization zap was produced.

AUD024   **INVALID CHECKSUM IN AUTHORIZATION ZAP**

This message indicates that the *SirAud* authorization zap was incorrectly entered. If the zap was manually entered, double check the contents from the original copy. If the zap was received electronically from Sirius Software, either it was modified during receipt or was incorrectly transmitted. You should double check your steps for receiving it or obtain a new zap from Technical Support.

AUD025   *JPATCH usage*

This indicates the offsets in the JPATCH CSECT which have been changed from the initial value of zero, which approximates the maintenance level of the executing load module. ***JPATCH usage*** can have one of the two following forms:

1. **NO JPATCH USED**, which indicates that the entire patch space is zero.

2. **JPATCH USAGE: *beg-end ...*** , which indicates the ranges (in hexadecimal) in the patch area which have been changed from zero.

AUD026   **INVALID VALUE OF** *param_name* **PARAMETER**

This indicates that you entered a program parameter to *SirAud*, and that the value specified for the parameter is not allowed. The name of the parameter in error is shown by *param_name*.

AUD027   **UNKNOWN SIRAUD PARAMETER:** *str*

This indicates that you incorrectly entered the program parameters to *SirAud; SirAud* was expecting the name of a parameter, but instead it found the string *str*, which is not the name of a parameter.

AUD028   *Debugging text*

This message should normally not appear; it is used under the direction of Sirius Software to obtain diagnostic information for *SirAud*.

AUD029   **NO REPORTS SELECTED**

This indicates that *SirAud* will not produce any information because the SYSPRINT file definition is missing and either:

1. You specified one or more of the "report selection" parameters in such a way that no reports are selected (for example, you specified NOPR as the only report selection parameter).

2. You did not specify any report selection parameters and the file definitions for all report files (PERFREPT, PROCREPT, etc.) are missing.

<hr>

APPENDIX B *Installation*

You can install *SirAud* from a product tape or from an object deck downloaded from the Sirius Software web site. Installation from tape is described in "MVS Installation" on page 33 and "CMS Installation" on page 32. Installation from the web is described in the following section, "Installation from the web".

All *SirAud* distributions come with *all* maintenance pre-applied: up to the time the tape was cut for tape installations, and up to the time the object deck was downloaded for web installations.

If you are a user of a previous version of *SirAud*, see the *SirAud Release Notes* for the version you are installing. This document is available in the Rocket Software web site Documentation Library (**http://docs.rocketsoftware.com/nxt/gateway.dll?f=templates$fn=default.htm)**. The *SirAud Release Notes* highlights changes in the new version and lists any compatibility issues with the previous version.

B.1   Installation from the web

You can download the *SirAud* object files from the Sirius Software web site (**http://sirius-software.com**). The download process requires a userid and password.

1. Click the `Support` navigation link to go to the Customer Service page (**http://sirius-software.com/support.html**), a public page that explains how to get the required userid and password and that also contains links to the various download pages.

2. Click the `Download object files` link to go to a protected page (**https://sirius-software.com/maint/objlist**) that contains a dynamically-generated list of the various Sirius products that you may download.

    The page also contains a `Click here` link that causes the page to be re-displayed with detailed download and installation instructions. These instructions supercede any information in this manual.

3. On the `Download object files` page, select the link to the currently available *SirAud* object file for the version you require. This file is used for all the versions of *Model 204* that are currently supported by *SirAud*.

B.2   CMS Installation

SirAud is distributed on a single, 9-track, 6250 BPI magnetic tape containing the distribution files in VMFPLC2 format. You will need approximately 30 4K-byte blocks of CMS disk space to load the files.

1.   Mount the installation tape on a tape unit with a virtual address of 181.

2.   Issue the following command to load the installation tape contents to disk.

```
VMFPLC2 LOAD * * A
```

3.   Define and format a minidisk to receive the Sirius product distribution tape.

   Note that all Sirius products you have ordered will be contained in a single file on the tape. Refer to the tapemap supplied with the distribution kit for the minimum number of blocks required to contain the installation files, or simply allocate a 3400 4K byte block minidisk, which is sufficient regardless of the number of products you are installing.

4.   Attach a tape drive at virtual address 181 to the owner of the installation minidisk defined in the previous step. Mount the product distribution tape on this tape drive.

   The product distribution tape consists of 1 physical files in VMFPLC2 DUMP format. The files contain the following items :

File Name	Contents
SIRAUD LOADLIB	The SIRAUD load module.
SIRAUD EXEC	EXEC to invoke *SirAud*. This contains all necessary GLOBALs, and all necessary FILEDEFs except for JOURNAL, required to run *SirAud* in a CMS environment.
SIRAUD NAMES	A file used by SIRAUD EXEC with the CMS NAMEFIND command; the principal purpose of SIRAUD NAMES is to minimize changes to SIRAUD EXEC from version to version, keeping most updates localized in SIRAUD NAMES.

   In addition, your tape will contain the *SirZap* facility, whose use is optional. Its installation and usage is described in the *SirZap User's Guide and Reference*.

5.   After unloading the *SirAud* load module, you can customize it to your site's needs, if necessary; see "Customization of Defaults" on page 35.

B.3   MVS Installation

SirAud is distributed on a single 9-track, 6250 BPI magnetic tape containing the following files in IEBCOPY format. Installation of *SirAud* consists of simply loading the distribution files to disk and modifying the sample JCL to conform to local standards.

The following JCL will load the installation tape files. Note that this step will also load any other Sirius products you've ordered. The space allocations should be sufficient regardless of which products will be installed.

```
//LOADSIR  JOB  0,CLASS=A,MSGCLASS=X,NOTIFY=ME
//*
//*        Load Sirius products from tape
//*
//LIB      EXEC PGM=IEBCOPY,REGION=0M
//T1       DD   UNIT=TAPE,LABEL=(1,SL),
//              DISP=(OLD,PASS),DSN=SIRIUS.LIB
//T2       DD   UNIT=TAPE,LABEL=(2,SL),
//              DISP=(OLD,PASS),DSN=SIRIUS.LOAD
//T3       DD   UNIT=TAPE,LABEL=(3,SL),
//              DISP=(OLD,PASS),DSN=SIRIUS.ULSPF
//*
//D1       DD   DISP=(,CATLG),SPACE=(CYL,(10,0,5),
//              DSN=SIRIUS.LIB,UNIT=SYSDA
//D2       DD   DISP=(,CATLG),SPACE=(CYL,(50,0,2),
//              DSN=SIRIUS.LOAD,UNIT=SYSDA
//D3       DD   DISP=(,CATLG),SPACE=(CYL,(25,0,2),
//              DSN=SIRIUS.ULSPF,UNIT=SYSDA
//SYSPRINT DD   SYSOUT=*
//SYSIN    DD   *
COPY INDD=T1,OUTDD=D1
COPY INDD=T2,OUTDD=D2
COPY INDD=T3,OUTDD=D3
/*
```

The tape contains three files, each in IEBCOPY format, but only the first two are used for *SirAud* installation. The relevant members for *SirAud* in these two files are described in the following list:

SIRIUS.LIB   Contains object files and JCL for linking and running *SirAud* under MVS.

SIRIUS.LOAD   Contains the SIRAUD load module.

In addition, your tape will contain the *SirZap* facility; use of *SirZap* is optional. *SirZap* usage and installation instructions are described in the *SirZap User's Guide and Reference*.

After unloading the *SirAud* load module, you can customize it to your site's needs, if necessary; see .

Installation

APPENDIX C   *Customization of Defaults*

You can change some of the *SirAud* defaults by applying zaps any time after you have installed the load module. This chapter contains the customizations you can perform.

C.1   ALLPR parameter customization

Apply the following zap to make the ALLPR parameter the default:

```
NAME SIRAUD JCUST
VER Ø8 ØØ
REP Ø8 8Ø    ALLPR is default
```

APPENDIX D *Date Processing*

This chapter notes there are no date processing issues for *SirAud*: *SirAud* does not produce any results that depend on the content of any data that may be date values.

Headers on pages or rows that occur on printed pages or displayed screens
0 generally have a full four-digit year format, although they may display dates with two-digit years in circumstances where the proper century can be inferred from the context.

Date Processing

Index

A
ALLPR parameter ... 7

F
FILERP parameter ... 10
FROM parameter ... 8

I
Installation ... 31
   Download from web ... 31

M
MAXBLK parameter ... 10

N
NAMEOFF parameter ... 10
NFILES parameter ... 10
NOFILERP parameter ... 10
NOPERFRP parameter ... 11
NOPR parameter ... 7
NOPROCRP parameter ... 11
NOSUBRP parameter ... 12
NOUTBLRP parameter ... 13
NOUTBLSM parameter ... 13
NPROCS parameter ... 10
NSUBS parameter ... 10
NUSUMM parameter ... 11
NUTBLS parameter ... 11

P
Parameters ... 7
PERFRP parameter ... 11
PROCRP parameter ... 11

S
SKIPBLK parameter ... 12
SUBRP parameter ... 12

T
TO parameter ... 12

U
UTBLRANG parameter ... 13
UTBLRP parameter ... 13
UTBLSM parameter ... 13

Index