CA JARS® Resource Accounting

Systems Programmer Guide

Release 12.7

This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the “Documentation”) is for your informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and may not be disclosed by you or used for any purpose other than as may be permitted in (i) a separate agreement between you and CA governing your use of the CA software to which the Documentation relates; or (ii) a separate confidentiality agreement between you and CA.

Notwithstanding the foregoing, if you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

CA Technologies Product References

This document references the following CA products:

- CA ACF2™.for z/OS
- CA Auditor for z/OS
- CA Common Services for z/OS
- CA MICS® Resource Management
- CA Service Desk
- CA SMF Director®
- CA Top Secret® for z/OS

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Documentation Changes

The following documentation updates have been made since the last release of this documentation:

Note: In PDF format, page references identify the first page of the topic in which a change was made. The actual change may appear on a later page.

- Added the Customization (see page 41) sections.

Contents

Chapter 4: Special Usage Considerations 61

Chapter 5: User Exits 77

Chapter 6: Utilities 105

Chapter 7: Using CA JARS with Special Environments 137

Chapter 8: CMF Processing 153

Appendix A: CA JARS Exits 173

Chapter 1: System Description

This guide serves as the primary reference for the systems programmer after CA JARS Resource Accounting has been installed. It is intended for the person responsible for implementing or extending the capabilities of this product through user exits. It also benefits the person(s) responsible for:

- capacity planning
- performance evaluation
- developing a chargeback system

CA JARS is comprised of a number of components, or subsystems, working together as a whole. The primary inputs to the system are:

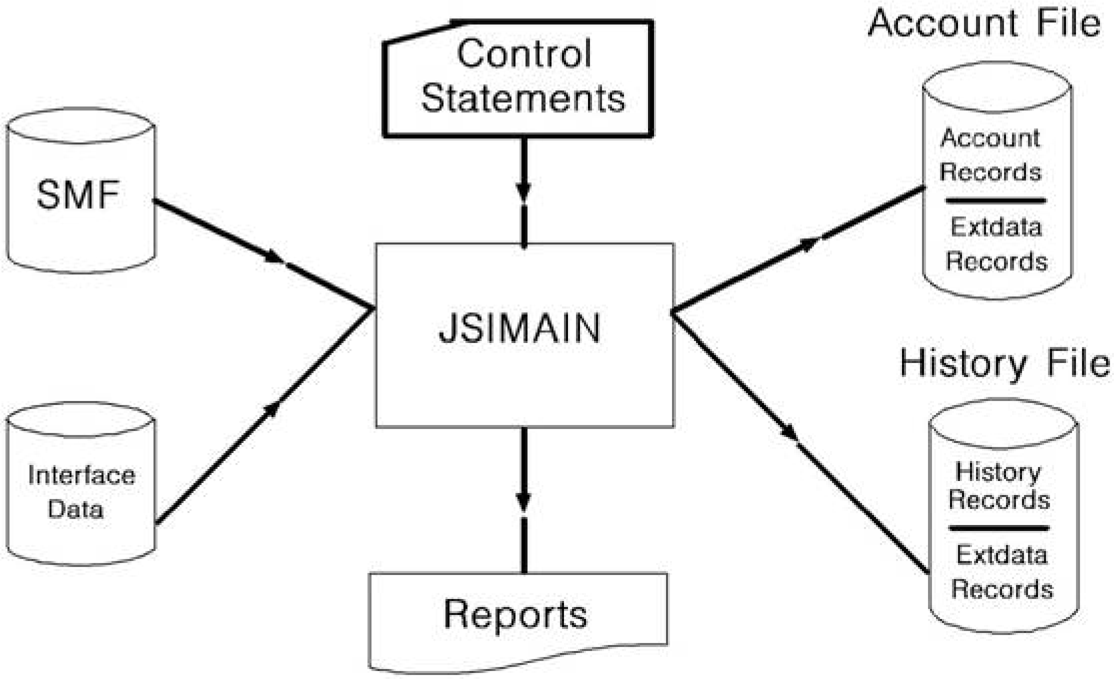
- SMF data
- interface data
- control statements

The primary outputs are:

- history records
- EXTDATA records
- reports

This section contains the following topics:

System Flow - CA JARS Subsystem



Operation of the JSIMAIN program is described in the *User Guide.* Various interfaces are described in the *Interfaces Guide* and the *User Guide.*

EXTDATA and Account/History Records

The difference between history records and account records is detailed in the chapter "Performance Considerations." In this section, account record refers to either account or history records.

The difference between EXTDATA and account records is really the difference between a physical and a logical record. SMF data is read by the JSIMAIN program, and, in the case of EXTDATA, is simply reformatted for later use by generalized report programs, or for direct input to the JARS/OLF component. In the case of account records, the SMF data is logically combined into job or step records. These account records contain a synthesis of information from many different SMF records. The EXTDATA record (with certain exceptions) contains information from a single SMF record. Therefore, the content of a complete file of account records approximates that of a complete file of EXTDATA records; however, the number and composition of the individual records are very different.

JSIMAIN Control Statements

The majority of control statements used in the JSIMAIN program pertain to account records, their creation, and reporting. The EXTDATA control statement, which controls the production of EXTDATA records, is the only statement pertaining to EXTDATA records.

JSIMAIN Reports

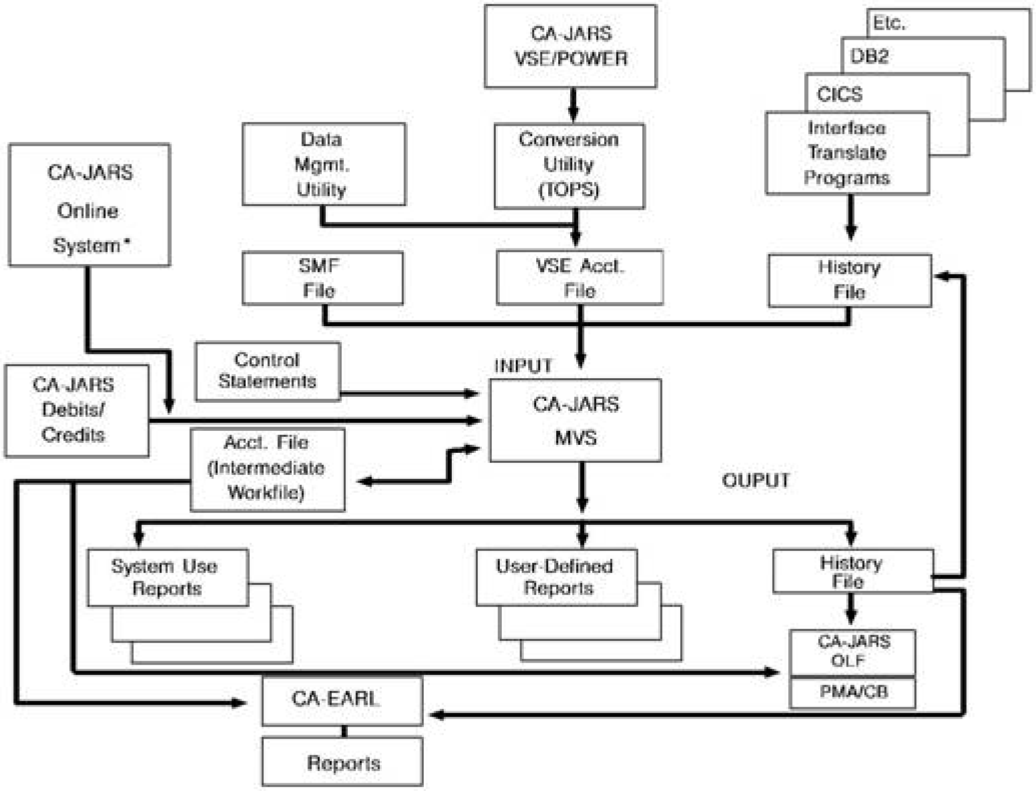
It should be noted that EXTDATA records are not used for JSIMAIN reporting purposes. JSIMAIN is simply a 'black box' for producing EXTDATA records. Reports, charges, etc., produced by JSIMAIN cannot access EXTDATA information. If EXTDATA records are requested by using the EXTDATA control statement, the account and history files produced by JSIMAIN contain the EXTDATA records requested.

Why Use EXTDATA Records?

The major reasons for creating EXTDATA records are to attain access to new metrics in SMF data for reporting and chargeback purposes, and for access to RMF data made available only through EXTDATA records.

The CA JARS account records are, by nature, of a fixed length. As additional information has been added to existing SMF records and new SMF records have been created, it has been a requirement for users that they have the ability to charge for these new items. By using EXTDATA records to include this new information, existing operation of the JSIMAIN program and its production of account records can remain constant. For the majority of users, the production of EXTDATA records can be omitted and the account records will suffice. However, for those whose requirements include these newer items, EXTDATA either alone, or in conjunction with the account records, provides an excellent solution.

Accessing EXTDATA Information



The CA Earl sample reports which illustrate the use of EXTDATA records for reporting are provided. Definitions and samples are also provided to illustrate the use of account records for reporting.

Definitions are provided for the CA Easytrieve language to allow users to quickly implement reporting using that language. However, any reporting program or user written program may access the account or EXTDATA records. Again, please note that the EXTDATA records and account/history records exist in the account and history files. Refer to the EXTDATA Reporting chapter in the *User Guide* for details on EXTDATA processing.

***** See the *Online System User Guide* for a detailed description of this feature.

SMF Record Usage

The following table lists the SMF record types used to create EXTDATA and account records:

SMF Type	Account/History Record	EXTDATA Record
0	N	Y
4	Y	N
5	Y	N
6	Y	Y
7	Y	Y
8	N	Y
9	N	Y
10	N	Y
11	N	Y
19	N	Y
25	Y	N
26	Y	N
30	Y	Y
30-1	Y	Y
30-2	Y	Y
30-3	Y	Y
30-4	Y	Y
30-5	Y	Y
32	N	Y
33	N	Y
39	N	Y
41	N	Y
42	N	Y
42-5	N	Y
42-6	N	Y
47	Y	Y

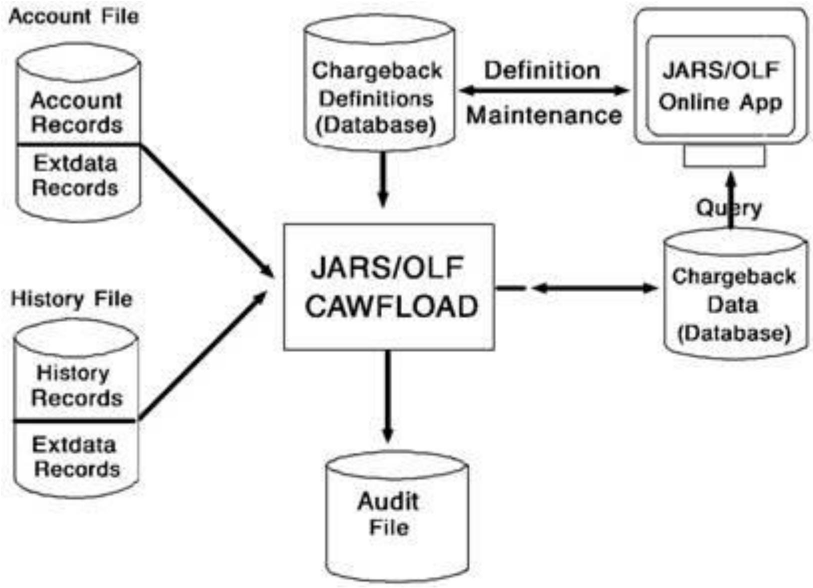
SMF Type	Account/History Record	EXTDATA Record
48	Y	Y
49	Y	Y
50	N	Y
52	Y	Y
53	Y	Y
54	Y	Y
57/59	N	Y
70	Y	Y
71	N	Y
72	N	Y
73	N	Y
74	N	Y
75	N	Y
77	N	Y
78	N	Y
88	N	Y
89	N	Y
92	N	Y
94	N	Y
101	N	Y
103	N	Y
110	N	Y
115	N	Y
116	N	Y
118	N	Y
119	N	Y
120	N	Y

As can be seen, a number of new SMF records are processed through EXTDATA records, allowing for more advanced resource utilization reporting than ever before.

In addition to providing extensive reporting capabilities, this product extends the power of chargeback to new heights with the inclusion of the JARS/OLF component. This component takes advantage of the most modern and user-friendly techniques to provide an easy to use and comprehensive chargeback system which allows users to take the next step from the more basic JSIMAIN chargeback methodology.

Note: For information regarding the usage and capabilities of JARS/OLF, please refer to the *JARS/OLF 1.0 User Guide.*

JARS/OLF Component



JARS/OLF provides a simple yet powerful way to create and maintain a centralized chargeback system. It also provides immediate access to charge information without the need for ad hoc reports to be run. The online application also supports advanced chargeback features such as:

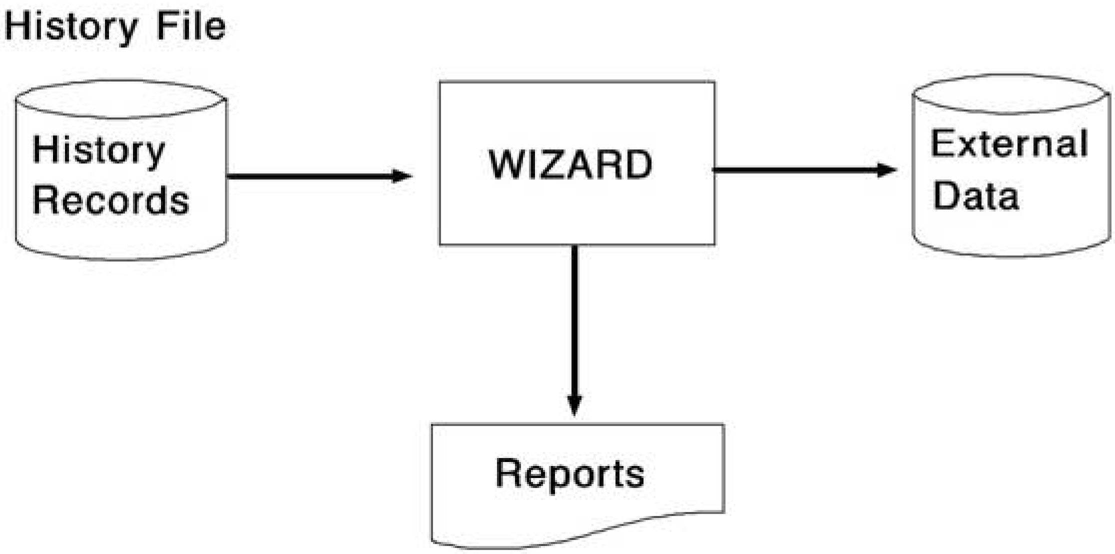
- ability to charge on any field
- multiple algorithms based on user-defined qualifications
- multiple rate sets
- online query (available to user as well)
- Forecasting
- rate determination
- support for natural business units

For more information, please refer to the *JARS/OLF 1.0 User Guide.*

Note: Both EXTDATA records and account records can be read simultaneously by the CAWFLOAD program. The production of EXTDATA records is not required to use the JARS/OLF component.

CA JARS Wizard

The Wizard Report Writer is provided for those users who have used it in past releases of CA JARS for upward compatibility. It may also prove to be a more comfortable tool for those users migrating from a CA JARS/VSE environment where CA JARS Wizard is the main reporting tool.



System Flowcharts for Writing Custom Exits

If you do not want to customize the product by writing a CA JARS exit, you should skip this section.

The system control data flow falls into five general phases:

- system command verification
- account file creation
- system use report generation
- report command verification
- user report generation

The table on the following page gives an overview of each phase. Following the table is a more detailed description of each phase.

Detailed specifications on the various exit points are described in the "User Exits" chapter.

Phase #	Description	Input DDs	Output DDs	Exits
Phase 1	System Command Verification	-CAIJSCIN	-CAIJSCT1	None
Phase 2	Account File Creation	-CAIJSSMF -CAIJSDOS -CAIJSAIP -CAIJSHST	-CAIJSACT -CAIJSNAP	Input Exit Account Exit
Phase 3	System Use Report Generation	-CAIJSACT	-CAIJSPRT	None
Phase 4	Report Command Verification	-CAIJSCT1	-CAIJSCT2	None
Phase 5	User Report Generation	-CAIJSCT2 -CAIJSACT	-HISTOUT -User Defined Reports	Exit 1 Exit 2 Exit 3

Of the five phases, three invoke the system sort: phases 2, 3 and 5. Phases 1 and 4 perform the primary function of command verification and have no requirement for sort. Where the sort is required, input is supplied by an appropriate E15 sort exit and disposed of (processed on the back end of sort) by an E35 sort exit. These exits process both the input and output files, thus supplying and disposing of the appropriate information. This architecture precludes the need for either a SORTIN or a SORTOUT data definition statement in the run stream.

Phase 1: System Command Verification

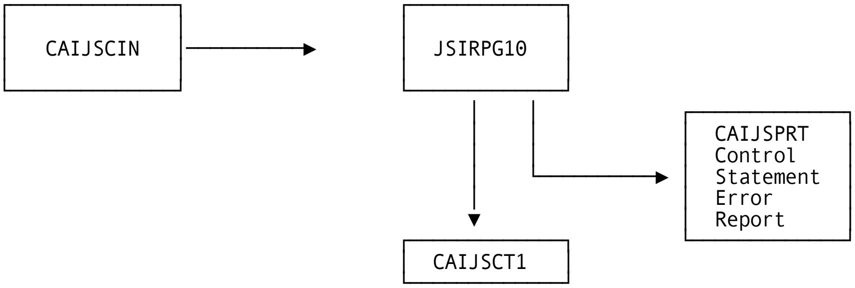
Input	Process	Output
CAIJSCIN -		

Input	Process	Output
Command Input Stream	Read Command Input Stream into the Output file: CAIJSCT1 Edit the System Commands: See Note 1 on "Special Notes". User Exits: None	- CAIJSCT1 - CAIJSPRT System Command Error Report

This first phase:

- reads in all the control statements from CAIJSCIN.
- writes to CAIJSCT1: editing system related control statements.
- uses the following programs: JSIRPG0 (JSIMAIN), JSIRPG1 (JSIRPG10).

System Command Verification



Phase 2: Account File Creation

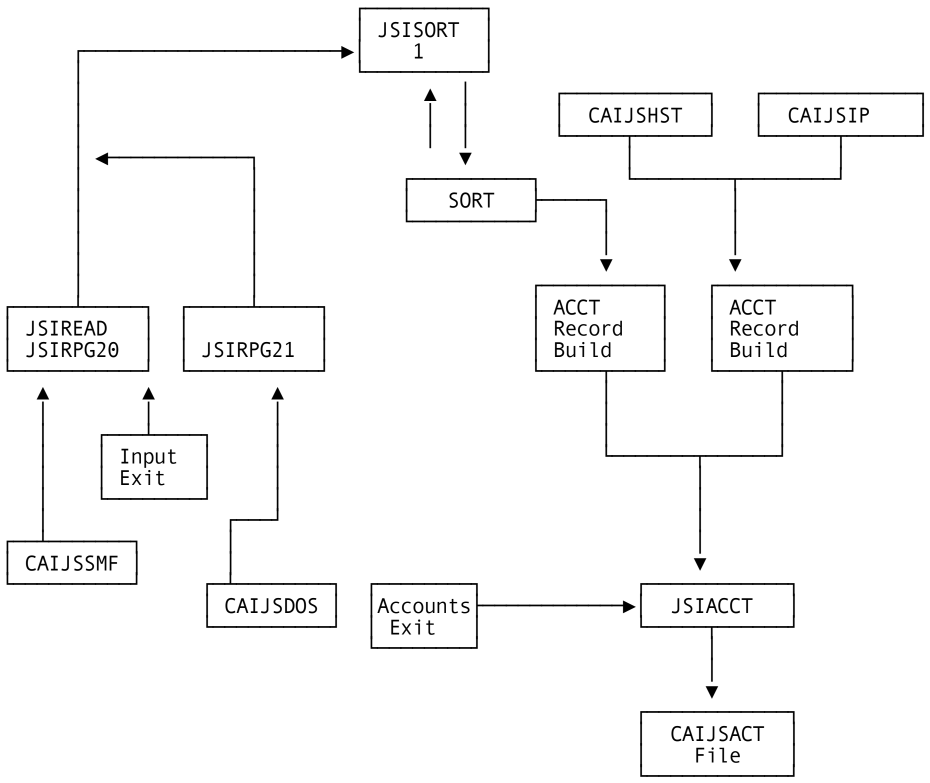
Input	Process	Output
CAIJSSMF CAIJSHST	Supply modules acting as an E15 sort exit. Reformat the SMF and DOS data into condensed records called SMF transmittal records. These are passed to sort. Certain EXTDATA records are written directly to the account file. See Note 2 on Special Notes.	
CAIJSAIP CAIJSHST	Disposal modules acting as an E35 sort exit create the account file from:	
	■ the sorted transmittal records ■ a previously created account file ■ a previously created history file	- CAIJSACT
User Exits:		

Input	Process	Output
	■ The INPUT Exit is invoked for every SMF record read. ■ The ACCOUNT Exit is invoked prior to writing the account record to the account file.	

In this second phase:

- If CAIJSSMF is selected:
  - SMF data is read in
  - Input Exit is called (if indicated on PARMS statement)
  - IPL messages and SMF summary is written
  - selection based on criteria date checking is performed
  - certain EXTDATA records are written directly to the account file
  - transmittal records are created from the SMF record types CA JARS uses. See Note 2 on Special Notes.
  - sorts data to pull together all records related to the same job execution
  - duplicate record checking is performed
  - account records are created
- If CAIJSDOS is selected:
  - reads DOS data
  - performs criteria date checking
  - creates transmittal records
  - performs duplicate record checking
  - creates account records
- If CAIJSHST and/or CAIJSAIP are selected:
  - reads history and/or CAIJSAIP data
  - performs criteria date checking
  - creates account records
- For all inputs:
  - Account Exit is invoked (if indicated on PARMS statement)
  - account records are written to the working data base (account file)
  - control statements: SELECT, CRITERIA, PARMS, CONFIG/CONFIGX

CAIJSACT File Creation



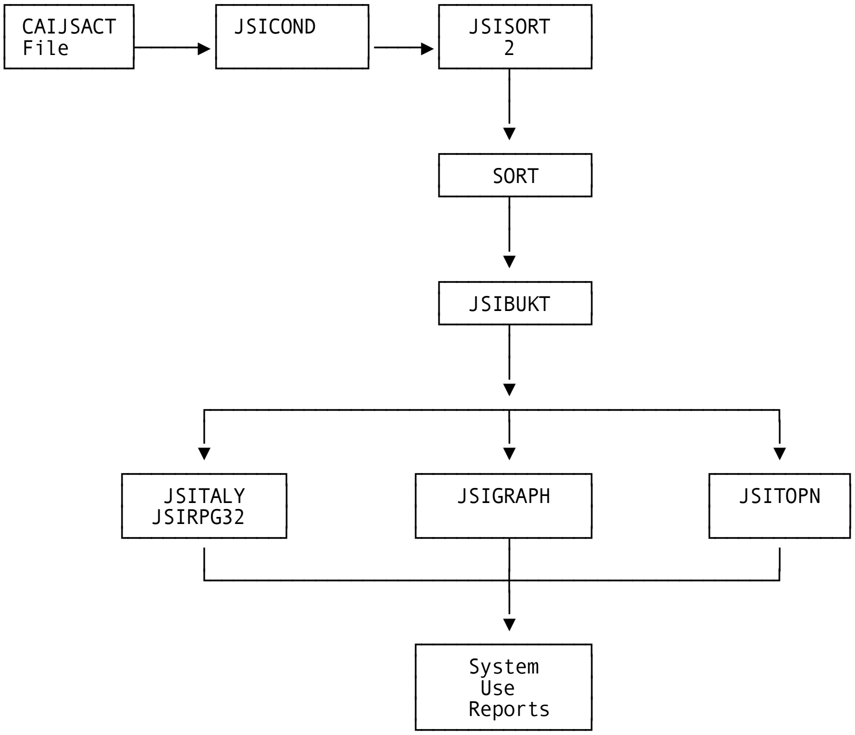
Phase 3: System Use Report Generation

Input	Process	Output
CAIJSACT	A supply module acting as an E15 sort exit reformats the account records into condensed records called SYSUSE transmittal records. These are passed to sort. See Note 3 at the end of this chapter.	
	Disposal modules acting as an E35 sort exit summarize the sorted transmittal records and create:	■ System Use Reports
	■ System Use Reports ■ Program Utilization and Job Utilization Graphs and Reports. User Exits: None	

This third phase:

- creates system use reports (if indicated on SELECT statement)
  - sorts SYSUSE transmittal records created from CAIJSSMF, CAIJSAIP or CAIJSHST (level 6 or 7)
  - sets up parameters, accumulators and table area
  - builds tables needed in:
    - Multiprogramming Analysis Report
    - Graphs and Summary Reports
    - Program/Job Reports
- uses programs: JSISORT3, JSICOND, JSIRPG30, JSITALY, JSIGRPH, JSITOPN, JSIRPG32, JSIBUKT, JSIRPG31, JSIRPG33, JSIRPG34
- control statements: SELECT, SYSUSE1, SYSUSE2

System Use Report Generation



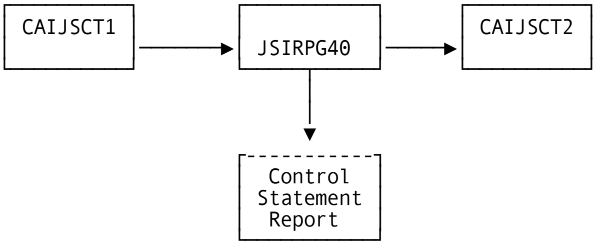
Phase 4: Report Command Verification

Input	Process	Output
CAIJSCT1	Reads and edits report command input stream	CAIJSCT2

This fourth phase:

- selects and edits report and global control statements for each report and writes to CAIJSCT2
- produces user-defined report control statement report
- uses files: CAIJSCT1, CAIJSCT2
- control statements: HEADER, TITLE, SORT, DISPLAY, DESCRIPT, GROUP, GROUPC, EDIT, RATE, TSORATE, PRIORITY, FORMRATE, RJERATE, DEVADDR, EXITS, CREDIT, DEBIT, BUDGET, STDFORM, STDPUNCH, STDPRINT, SNARATE, APPCRATE, DEVNMB

Report Command Verification



Phase 5: User Report Generation

Input	Process	Output
CAIJSCT2	Control tables are built based on the following control statements: ■ RATE ■ TSORATE/RJERATE/SNARATE/APPCRATE ■ GROUP/GROUPC ■ PRIORITY ■ DEVADDR/DEVNMB Account records are created from CREDIT, DEBIT and BUDGET statements. An account record is read; selection and rejection (grouping) is performed. Rates are applied.	
	Detail history file (job, level 6 or step, level 7) or summary history file (levels 1-5) is created.	HISTOUT (User-Defined ddname)
	Output report is printed.	User-Defined Reports
	Phases 4 and 5 are executed once for each of the 15 possible user-defined reports. User Exits: ■ Exit 1 is called prior to rate application. ■ Exit 2 is called after rate application. ■ Exit 3 is called just prior to printing the output line.	

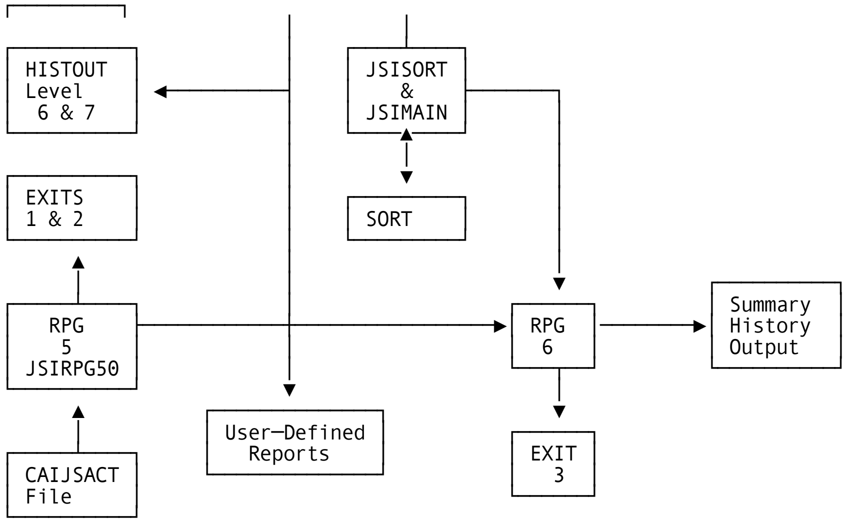
Note: EXTDATA records are not available for reporting or chargeback.

This fifth phase:

- builds control tables for the following control statements:
  - RATE, TSORATE, RJERATE, SNARATE, APPCRATE
  - GROUP, GROUPC
  - PRIORITY
  - DEVADDR, DEVNMB

- creates account records (one-step job records) for CREDIT, DEBIT, and BUDGET statements
- applies grouping selection/rejection logic to all records
- EXIT1 is invoked (if indicated on EXITS statement) so user may modify, delete or apply unique rates to specific account records
- applies rates
- EXIT2 is invoked (if indicated on EXITS statement) to delete, modify, or apply unique rates to specific account records
- DEVADDR/DEVNMB selection/rejection logic is applied
- creates a level-6 or -7 History File (if indicated on SORT statement) for job or job and step level detail, which .us on includes .us off any EXTDATA records requested
- sorts and summarizes account records
- creates level 1-5 Summary Files (if indicated on SORT statement), which does .us on not include .us off EXTDATA records
- creates Output Data Elements Table
- EXIT3 is invoked (if indicated on EXITS statement), to delete or modify records in the Output Data Elements Table
- creates print lines from the Output Data Elements Table and report is generated
- uses programs: JSIRPG5, JSISORT3 (JSIMAIN), JSIRPG6, JSIRPG51
- uses files: CAIJSACT, HISTOUT
- control statements: CREDIT, DEBIT, BUDGET, GROUP, GROUPC, EXIT, RATE, TSORATE, FORMRATE, RJERATE, PRIORITY, SORT, DEVADDR, DEVNMB DISPLAY, DESCRIPT, EDIT, TITLE, HEADER
- Phases 4 and 5 are executed once for each of the 15 possible user-defined reports

User Report Generation



Special Notes

Note 1

System *(global)* commands are applicable to every user report in a CA JARS run and are listed here:

- RATE
- FORMRATE
- RJERATE
- SNARATE
- APPCRATE
- DEVADDR
- DEVNMBR
- EDIT
- DESCRIPT
- CREDIT
- DEBIT
- BUDGET
- EXITS
- TITLE

- TSORATE
- VMRATE
- CONFIG
- CONFIGX
- STDPUNCH
- STDFORM
- SELECT
- CRITERIA
- PARMS
- SYSUSE1
- SYSUSE2
- OPTION
- EXTDATA
- STDPRINT
- EXTDATA

Note 2

This product uses a variety of the SMF record types to produce the account records. The following is a list of the record types used:

SMF Type 30	Common address space work record
Subtype 1	Job initiation
Subtype 2	Interval End
Subtype 3	Step/Session Termination
Subtype 4	Step/Session End
Subtype 5	Job/Session End

Other batch job record types that may be used:

Type 26	JES2/3 Job Purge
Type 6	JES2/3 Output Writer

For APPC:

Type 33	APPC/MVS TP Accounting

For RJE jobs the record types used are:

Type 47	RJE JES2/3 Signon
Type 48	RJE JES2/3 Signoff
Type 49	RJE JES2/3 Integrity
Type 52	JES2 SNA Signon
Type 53	JES2 SNA Signoff
Type 54	JES2 SNA Integrity

System record types used are:

Type 0	IPL
Type 7	Data Lost
70	CPU Activity

Note 3

The creation of the SYSUSE transmittal records is subject to the following rules:

1. If the passed account record contains data reflecting activity *days* (delimited by shift 1 start time), a SYSUSE transmittal record is created for each of those *days* that passed the criteria test.

2. Account records with no steps do not produce SYSUSE transmittal records. This event occurs when all records for the job have been lost or discarded with the exception of any type 6 SMF records.

3. Of the 13 possible record types on the account file, only the following five are used to generate SYSUSE reports:

   Type

S	SMF Job/Step
T	TSO
@	IPL
#	WAIT TIME
?	SMF Lost Data

For RJE jobs the record types used are:

Type	
47/52	RJE JES2/3 Signon
48/53	RJE JES2/3 Signoff
49/54	RJE JES2/3 Integrity

System record types used are:

Type

0 IPL

Data Lost

CPU Activity

Program Summary Table

CSECT	MODULE	Process
JSIRPG0	(JSIMAIN)	Initiates each phase of the execution control module.
JSIRPG1		Edits system control statements.
JSISORT		Sort control module. Passes records to user installation sort program.
JSIREAD	(JSIRPG20)	SMF file processing module. SNAPs dropped records or blocks. Produces SMF record type summary and prints IPL messages.
JSISPAN		Deblocks and builds records from segments (SMF data) as necessary. Performs validation of block, record and segment descriptor words. Invalid control information causes segment or block to be dropped.
JSIRPG2		DOS account file processing module. Edits, then reformats records into transmittal records and passes them to JSISORT.
JSIRP3S	(JSIRPG22)	SMF transmittal record processing module. Creates job and step account records from transmittal records. Builds system related records.

CSECT	MODULE	Process
JSIRPG3	(JSIRPG23)	DOS transmittal, old account, history input record processing module. Builds account records. Prior release account records are not supported. Passes valid records to JSISORT.
JSIACCT		Write account/summary files. Controls access and writes to account and summary files.
JSICOND	(JSIRPG30)	Builds SYSUSE transmittal records from account records for: &bullet. Execution of Job and/or Step &bullet. Turnaround of Job &bullet. Wait Time Statistics
JSIBUKT	(JSIRPG31)	Supervises the production of graphs and TOPN reports. Allocates table areas for JSITALY.
JSITALY	(JSIRPG32)	Summarizes (tallies) by day, device, time and/or program/job (for system use reports only).
JSIGRPH	(JSIRPG33)	Produces graphs and/or Computer Utilization Summary.
JSITOPN	(JSIRPG34)	Generates Job/Program Activity Analysis Report.
JSIRPG4	(JSIRPG40)	Edits report control statements.
JSIRPG5	(JSIRPG50)	Processes input for user-defined reports, applies rates, and creates detail-level history files.
JSIRPG6		Prints reports and builds summary history files.

Chapter 2: SMF Extensions

The System Management Facility Extensions (SMF/E) enhance z/OS by providing realtime accounting and performance reports in each JOB LOG and SYSMSG output listing, and at the end of each TSO session.

This chapter discusses the following areas of consideration concerning SMF/E:

- features and benefits
- report output
- activating, deactivating, and dynamic control
- customization
- the chargeback algorithms used
- system requirements

After you have properly established your rates and during the initial stages of testing SMF/E at your installation, you may want to ensure that the output only appears in your own test jobs. You can accomplish this using the OUT(IVP) and TSO parameters discussed later in this chapter.

This section contains the following topics:

Features and Benefits

The major features of SMF/E are:

- Processing Summary Report: Return codes and estimated job/step charges printed in the JOB LOG output for each batch job.
- Step Summary Report: Estimated step charge, CPU timings, storage utilization, and EXCP counts by ddname are printed in the SYSMSG listing for each step.
- Job Summary Report: Estimated total job charge, CPU timings, and summary I/O statistics printed in SYSMSG for the job.
- TSO Session Utilization Summary Report: Estimated session charge, CPU timings, and summary I/O statistics are displayed on the terminal when you LOGOFF a TSO session.
- Batch compatibility: SMF/E can be customized to provide near uniform charging between batch CA JARS and SMF/E for those statistics available at step and job termination.

When using SMF/E, the major benefits of these features include the following:

- Provide full function accounting at the step level and job status.
- Provide utilization statistics that can be the basis for program tuning and debugging.
- Reduce the need for user-written and user-supported system exits.
- Provide for a controlled implementation through use of the IVP option.
- Increase user awareness of the cost of their individual job executions.
- Locate batch jobs that are excessive resource consumers without requiring a batch CA JARS run.
- Installation Verification Procedure (IVP): This startup option provides a mechanism to allow a controlled group, such as the systems programming staff, to receive outputs from SMF/E while the remaining users are ignored by SMF/E. This allows a controlled phase-in period an opportunity to experiment with various rates and other chargeback factors.

Once you have properly customized your rate table, you want to notify all affected computer users of the new facility to avoid surprising them. This helps ensure greater acceptance of your chargeback system.

Report Output

SMF/E provides the following output:

- JOB LOG Output Messages
- SYSMSG Step Detail Report
- SYSMSG Job Summary Report
- TSO Session Utilization Summary

The JOB LOG Summary Report is an overview of the job and job step resource completion status and cost. One line per job step is written to the JOB LOG data set, and optionally to the SYSLOG data set. The following is a sample JOB LOG Summary Report. The report also includes any other JOB LOG output generated by other software vendors' products.

Note that this output can show the complete status of a job--the programs involved, their relative contribution to costs, and important condition codes. Whether viewed as the first page of a hardcopy listing or the first screen of an SDSF display, it provides an important consolidation of step and job status.

```
JES2 JOB LOG
                         J E S 2  J O B  L O G  --  S Y S T E M  X A D 1  --  N O D E  M V S

 09.50.12 JOB 4757  $HASP373 JRC0TAPE STARTED - INIT  9 - CLASS T - SYS XAD1
 09.50.12 JOB 4757  IEF403I JRC0TAPE - STARTED - TIME=09.50.12
 09.50.14 JOB 4757  CAJR250I STEPNAME STEP   PGM=   CCODE  EST-COST EXCPS     ELAPSED     TOT-CPU PAGE-IN PAGE-OT SWAP-IN SWAP-OT
 09.50.14 JOB 4757  CAJR251I FILE1CPY    1 IEBGENER 0000      $ .15    12 00.00.02.42 00:00:00.33       0       0       0       0
 09.50.15 JOB 4757 *IEF233A M 284,PRIVAT,SL,JRC0TAPE,FILE1,SMPMCS
 09.51.08 JOB 4757  IEC705I TAPE ON 284,100715,SL,1600 BPI,JRC0TAPE,FILE1,SMPMCS
 09.51.09 JOB 4757  CAJR251I FILE1       2 IEBGENER 0000      $ .23    25 00.00.54.25 00:00:00.38       2       0       0       0
 09.51.10 JOB 4757  CAJR251I FILE1PRT    3 IEBGENER 0000      $ .14    13 00.00.00.96 00:00:00.29       0       0       0       0
 09.52.02 JOB 4757  CAJR251I UNLOAD      4 IEBCOPY  0000     $12.72  2033 00.00.50.88 00:00:05.23       0       0       0       0
 09.52.04 JOB 4757  CAJR251I FILE11      5 IEBGENER 0000      $ .09     3 00.00.01.74 00:00:00.30       0       0       0       0
 09.52.13 JOB 4757  CAJR251I FILE12      6 IEBGENER 0000      $2.37   413 00.00.08.72 00:00:01.07       0       0       0       0
 09.52.33 JOB 4757  CAJR251I DSB         7 IEBCOPY  0000      $2.17   308 00.00.19.57 00:00:01.31       1       0       0       0
 09.52.33 JOB 4757  IEF234E K 284,100715,PVT,JRC0TAPE
 09.52.34 JOB 4757  IEF404I JRC0TAPE - ENDED - TIME=09.52.34
 09.52.34 JOB 4757  CAJR252I JOB ENDED. TOTAL EST-COST       $17.87    TOTAL CPU TIME 00:00:08.91
 09.52.34 JOB 4757  $HASP395 JRC0TAPE ENDED
```

The SYSMSG Step Detail Report has two formats--one of which is be used. The difference between the two is primarily one of formatting. Which report format is printed is specified at SMF/E initialization and can be overridden dynamically in each individual job execution under user control. See the section in this chapter called Installing, Enabling, Disabling, Reinitializing and Dynamic Control for an explanation of specifying the report types. Regardless of which format is chosen, the report is produced at the end of each job step.

The following is a sample of the first format of the Step Detail Report. It is produced by specifying the STEPP option. STEPP is the default option for SMF/E reporting.

```
STEP DETAIL REPORT (first format)

 ******************************************************************************
 * CA JARS r12          JOBNAME  JRC0TAPE                 CAIJR300 *
 * Resource Accounting SP0          STEPNAME  UNLOAD  05 AUG 1998 09.51.11.65 *
 *                               STEP NO.  4                                  *
 * ESTIMATED CHARGES:             PROGRAM  IEBCOPY        ELAPSED 00.00.50.88 *
 *  PROCESSOR CHARGE     $2.56  COND CODE  0000           TCB-CPU 00.00.03.77 *
 *        I/O CHARGE    $10.16 CORE ALLOC     10184K      SRB-CPU 00.00.01.46 *
 *      TOTAL CHARGE    $12.72 CORE  USED       768K    TOTAL-CPU 00.00.05.23 *
 *                                                                            *
 * ddname  ADDR DEVICE  EXCPS   CHARGE  ddname   ADDR DEVICE  EXCPS   CHARGE  *
 * I2      544  DISK       66   $ 0.33  SYS00307 544  DISK        0   $ 0.00  *
 * I3      544  DISK       55   $ 0.27  I4       546  DISK      602   $ 3.01  *
 * I5      546  DISK       24   $ 0.12  I6       541  DISK        8   $ 0.04  *
 * I7      543  DISK        5   $ 0.02  I8       543  DISK      593   $ 2.96  *
 * I9      44C  DISK       23   $ 0.11  I10      544  DISK        4   $ 0.02  *
 * O2      284  TAPE       73   $ 0.36  O3       284  TAPE      239   $ 1.19  *
 * O4      284  TAPE      220   $ 1.10  O5       284  TAPE        8   $ 0.04  *
 * O6      284  TAPE        4   $ 0.02  O7       284  TAPE       21   $ 0.10  *
 * O8      284  TAPE       74   $ 0.37  O9       284  TAPE        8   $ 0.04  *
 * O10     284  TAPE        6   $ 0.03  SYS00306 544  DISK        0   $ 0.00  *
 ******************************************************************************
```

The following is a sample of the second format of the Step Detail Report. It is produced by specifying the STEP option. STEP is an optional output format that fits within a 72 byte window for CRT viewing under some output processors. Its format is more lengthy than the STEPP format. For this reason the STEPP option may be somewhat more appropriate in an interactive environment where response time is an issue.

```
                                                                       PAGE 3

STEP LEVEL DETAIL REPORT (second format)
************************************************************************
* CA JARS r12                                       CAIJR200 *
* Resource Accounting SP0              05 AUG 1998 20.46.55.32         *
*                                                                      *
*         JOB NAME: JARIVP     STEP NAME: JARS       STEP NUMBER:   1  *
*                                                                      *
*     PROGRAM NAME: JSIMAIN    COND CODE: 0004                         *
*                                                                      *
* STORAGE & PAGING:                                                    *
*    PAGE-IN COUNT:       4    PAGE-OUT COUNT:       0                 *
*    SWAP-IN COUNT:       0    SWAP-OUT COUNT:       0                 *
*   CORE ALLOCATED:   10388K   CORE USED:          756K                *
*                                                                      *
* TIMES:                                                               *
*   ELAPSED   TIME:   00.01.06.00                                      *
*   TCB CPU   TIME:   00.00.11.16                                      *
*   SRB CPU   TIME:   00.00.01.12                                      *
*   TOTAL CPU TIME:   00.00.12.28                                      *
*                                                                      *
* ESTIMATED CHARGES:                                                   *
*     PROCESSOR CHARGE =             $5.99                             *
*           I/O CHARGE =             $8.46                             *
*         TOTAL CHARGE =            $14.45                             *
*                                                                      *
* I/O STATISTICS:                                                      *
*         ddname    ADDR  DEVICE    EXCPS            CHARGE            *
*         STEPLIB    544  DISK       401             $ 2.00            *
*         SYS00463   544  DISK         0             $ 0.00            *
*         SORTLIB    170  DISK         0             $ 0.00            *
*         SORTWK01   177  DISK         0             $ 0.00            *
*         SORTWK02   177  DISK         0             $ 0.00            *
*         SORTWK03   177  DISK         0             $ 0.00            *
*         SORTWK04   177  DISK         0             $ 0.00            *
*         SORTWK05   177  DISK         0             $ 0.00            *
*         SORTWK06   177  DISK         0             $ 0.00            *
*         CAIJSCT2   177  DISK        68             $ 0.34            *
*         CAIJSCT1   177  DISK      1035             $ 5.17            *
*         CAIJSACT   177  DISK        29             $ 0.14            *
*         CAIJSSMF   172  DISK       159             $ 0.79            *
*         SYS00465   449  DISK         0             $ 0.00            *
*         SYS00462   544  DISK         0             $ 0.00            *
*         SYS00464   449  DISK         0             $ 0.00            *
*                                                                      *
************************************************************************
```

The SYSMSG Job Summary Report has two formats--one of which is used. The difference between the two is primarily one of formatting. Which report format is printed is specified at SMF/E initialization and can be overridden dynamically in each individual job execution under user control. Regardless of which format is chosen, the report is produced at the end of each job execution.

The following is a sample of the first format of the Job Summary Report. It is produced by specifying the JOBP option.

```
                                                                       PAGE 4

Job SUMMARY REPORT (First Format)

******************************************************************************
* CA JARS r12         JOBNAME  JRC0TAPE                   CAIJR300 *
* Resource Accounting SP0          SYSTEM ID  XAD1   05 AUG 1998 09.50.12.30 *
*                             JOB CLASS  T                                   *
* ESTIMATED CHARGES:           PRIORITY¦ 00              ELAPSED 00.02.18.54 *
*  PROCESSOR CHARGE    $3.87                             TCB-CPU 00.00.06.84 *
*        I/O CHARGE   $14.00 CORE ALLOC     10188K       SRB-CPU 00.00.02.07 *
*      TOTAL CHARGE   $17.87 CORE  USED       768K     TOTAL-CPU 00.00.08.91 *
*                                                                            *
******************************************************************************
```

The following is a sample of the second format of the Job Summary Report. It is produced by specifying the JOB option. JOB is an optional output format that fits within a 72 byte window for CRT viewing under some output processors. Its format is more lengthy than the JOBP format. For this reason the JOBP option may be somewhat more appropriate in an interactive environment where response time is an issue.

```
                                                                       PAGE 5

JOB SUMMARY REPORT (second format)

************************************************************************
* CA JARS r12                                       CAIJR200 *
* Resource Accounting SP0                   05 AUG 1998 20.46.55.32    *
*                                                                      *
*         JOB NAME: JARIVP                            SYSTEM ID: XAD1  *
*                                                                      *
*        JOB CLASS: A                JOB PRIORITY: 00                  *
*                                                                      *
* STORAGE & PAGING:                                                    *
*    PAGE-IN COUNT:       4    PAGE-OUT COUNT:       0                 *
*    SWAP-IN COUNT:       0    SWAP-OUT COUNT:       0                 *
*   CORE ALLOCATED:   10388K   CORE USED:          756K                *
*                                                                      *
* TIMES:                                                               *
*   ELAPSED   TIME:   00.01.06.24                                      *
*   TCB CPU   TIME:   00.00.11.16                                      *
*   SRB CPU   TIME:   00.00.01.12                                      *
*   TOTAL CPU TIME:   00.00.12.28                                      *
*                                                                      *
* ESTIMATED CHARGES:                                                   *
*     PROCESSOR CHARGE =             $5.99                             *
*           I/O CHARGE =             $8.46                             *
*         TOTAL CHARGE =            $14.45                             *
*                                                                      *
* I/O STATISTICS:                                                      *
*                         DEVICE    EXCPS            CHARGE            *
*                         TAPE          0            $ 0.00            *
*                         DISK       1692            $ 8.46            *
*                                                                      *
*                                                                      *
************************************************************************
```

The following is a sample of the TSO Session Utilization Summary. It is produced by specifying YES on the TSO SMF/E initialization parameter.

```
                                                                       PAGE 6

TSO SESSION UTILIZATION SUMMARY
*****************************************************************************
* CA JARS r12           USERID TSOGXB                    CAIJR400 *
* Resource Accounting SP0         PROCNAME $TSOGXB  28 AUG 1998 09.46.05.49 *
*  PROCESSOR CHARGE    $23.64                            ACTIVE 00.01.00.37 *
*    CONNECT CHARGE     $2.44      TPUTS 91             CONNECT 00.29.14.50 *
*        I/O CHARGE     $ .31      TGETS 42             TCB-CPU 00.00.11.43 *
*        U/R CHARGE     $ .13 CORE ALLOC       404K     SRB-CPU 00.00.01.32 *
*      TOTAL CHARGE    $26.52 CORE  USED       164K   TOTAL-CPU 00.00.12.75 *
*                                                                           *
*                        DEVICE    EXCPS          CHARGE                    *
*                        TAPE          0          $ 0.00                    *
*                        DISK         62          $ 0.31                    *
*                                                                           *
*****************************************************************************
```

Installing, Enabling, Disabling, Reinitializing and Dynamic Control

This section documents the installation, enabling, disabling, reinitializing, and dynamic control features of SMF/E. Please note that *enable* means to initialize startup or activate the product and conversely *disable* simply means to terminate, shutdown or deactivate the product. These words are used interchangeably in this text. *Reinitialization* means that a new copy of the SMF/E processing module is loaded into memory. Use this option to apply or back out maintenance. SMF/E uses the CA Common Services for z/OS Services Resource Initialization Manager (CAIRIM) to prepare your operating system environment. Executing CAIRIM does **not** *enable* SMF/E. The enabling of SMF/E is discussed later in this chapter.

The CAIRIM PARM required for the CA JARS product family is:

```
PRODUCT (CA JARS)   VERSION (JRC6)   INIT (JARSINIT)
```

To reduce required resources in an environment where the CA JARS Resource Management for CICS product is not installed, use this statement instead:

```
PRODUCT (CA JARS)   VERSION (JRC6)   INIT (JARSINIT)  PARM (,JARS)
```

When the PARM specified at the end of the statement is passed to the JARSINIT program JARS, installation actions specific to CA JARS Resource Management for CICS are not performed. This reduces the resources required for installation in an environment without CA JARS Resource Management for CICS. If necessary, a subsequent CAIRIM initialization for CA JARS Resource Management for CICS will install its components.

The following is a sample job that can be used to execute CAIRIM and prepare the system for the enabling of SMF/E:

```
//CAIRIM    EXEC PGM=CAIRIM
//PARMLIB DD *
  PRODUCT(CA JARS) VERSION(JRC6) INIT(JARSINIT) PARM(,JARS)
//*
```

The CAIRIM routine is intended to be invoked as a stand-alone started task that can be optionally defined as a z/OS subsystem. See your CA Common Services for z/OS documentation for more information. The CAIRIM routine can also be used to reinitialize the global environment in order to apply maintenance. Once maintenance has been applied, run the following sample job to dynamically reinitialize the global environment:

```
//CAIRIM    EXEC    PGM=CAIRIM
//PARMLIB DD *
  PRODUCT(CA JARS) VERSION(JRC6) INIT(JARSINIT) PARM(REINIT,JARS)
//*
```

Specifying JARS as the second parameter (PARM(,JARS) and PARM(REINIT,JARS) in the sample code above) tells JARSINIT to only install the components required by the CA JARS Resource Accounting product. This reduces common storage use.

SMF/E is enabled and disabled by running a batch job, examples of which follow this discussion. The SMF/E control program accepts startup and shutdown specifications through the program PARM field. The PARM options are:

```
PARM=('LOAD(customization-table-name)',
      'ENABLE(exit-name)',
      'DISABLE(exit-name)',
      'REINIT(exit-name)',
      'OUT(default-output-format)',
      'ACCT(0|n)',
      'GNRC(YES|NO)',
      'TSO(YES|NO|userid)',
      'HCPY(NO|YES)',
```

where:

LOAD(customization-table-name) - required parameter, no default
Indicates the load module name of the customization table to be loaded at system activation. If this is specified while SMF/E is active, the table is reloaded and the new table is used.

ENABLE(exit-name) or DISABLE(exit-name) - required parameter, no default
Currently, the exit name must be JARSTATS since this is the only routine implemented. When ENABLE (JARSTATS) is specified and this startup run is completed, the next address space going through step-termination receives SMF/E output in accordance with these startup options. The ENABLE process is a very quick operation that can facilitate the experimentation and benchmarking of various customization tables.

When DISABLE is specified and the run is complete, the next address space going through step termination does not receive SMF/E output. When REINIT is specified, a new copy of the SMF/E processing module is loaded in the system. As in the previous cases, the new maintenance is executed after the run ends by the next address space going through step termination. Use the REINIT option only to apply or back out maintenance to the SMF/E processing module.

Note: This option should only be used for maintenance changes. Feature and rate table changes should be made with the ENABLE option.

OUT(output-format)
Indicates the report types that are produced in output listings. There are six different formats. The formats are listed below from most to least detailed. Each specification coded implies the inclusion of all report levels for which less detail is available.

For instance, requesting a STEP report implies that you will receive the JOB and SYSMSG reports.

- STEPP - Step Detail and Job Summary Reports (compressed format - default)
- STEP - Step Detail and Job Summary Reports (expanded format)
- JOBP - Job Summary Report (compressed format)
- JOB - Job Summary Report (expanded format)
- NONE - JOB LOG Summary Report only
- IVP - used during installation verification or testing to limit output to only those jobs that request output via a valid job card ACCT code. Specifying IVP in the "nth" ACCT code position produces output like STEPP, but it includes RATE and FACTORS.

ACCT(0|n)
Indicates the positional JOB card accounting field to be used for overriding the default output format. Individual users can override the system startup OUT parameter in this JOB card accounting field. The default is 0.

GNRC(YES|NO)
This option determines how generic or specific the classification of device types is in the output formats. GNRC(YES) indicates that the data collection routine for SMF/E determines the device type from the UCB attributes in the type 30 record before using the supplied CONFIG specification in the customization table. GNRC(NO) indicates that only the CONFIG table is used to classify device types.

TSO(YES|NO|userid)
TSO(YES) activates TSO session accounting for all TSO users. TSO(NO) deactivates the feature. TSO(userid) can be used to test TSO session accounting during the installation verification process. The indicated TSO user is the only one to receive the TSO Session Utilization Summary.

HCPY(YES|NO)
HCPY(NO), the default, prevents all WTO messages issued by SMF/E from being written to the System Log (SYSLOG). HCPY(YES) allows all Write-To-Operator (WTO) messages, issued from SMF/E to be written to the SYSLOG. Exceptions: for those jobs (notably started tasks) that have no RPL for the Job Log data set, SMF/E messages appear in the SYSLOG prefixed by message number IEF170I.

Note that the WTO messages are issued by SMF/E with ROUTCDE=11, which may show up on various system consoles. The value for HCPY controls SYSLOG only and other means are available to suppress or allow output to the consoles.

The following is a sample procedure you can use to activate SMF/E:

```
//SMFE  PROC RT=JSYJ2000,          SMF/E RATE TABLE
//          X=JARSTATS,            EXIT TO ENABLE (JARSTATS)
//          O=STEPP,               OUTPUT FORMAT REQUESTED
//          G=YES,                 GENERIC DEVICE ACCOUNTING
//          A=4,                   JOBCARD ACCT'G DATA PARM POSITION
//          T=YES
//SMFE  EXEC PGM=JSYJ1000,
// PARM=('LOAD(&RT.)','ENABLE(&X.)','GNRC(&G.)','OUT(&O.)',
//  'ACCT(&A.)', 'TSO(&T.)')
//SYSMDUMP DD  DISP=SHR,DSN=your.dsname
//CAIJRPR  DD  SYSOUT=*
//   PEND
```

Once you have completed customization of SMF/E, this procedure should be executed during your system IPL. SMF/E can be reactivated at any time to change any of the startup parameters.

SMF/E provides a facility that allows informed, individual users to choose the report types desired in each job execution by supplying the desired output format in the ACCT field defined by the systems programmer at SMF/E startup. Assuming SMF/E was enabled with ACCT(4) and OUT(STEP), the following JOB card eliminates step reporting and produce job and JOB LOG reports.

```
//GENER   JOB  (123,521,BIN4,JOB),CLASS=A,MSGCLASS=Z
```

The user may supply any of the keywords that are valid for the OUT parameter in the 4th positional ACCT field for this single JOB execution.

The following matrix shows the type of output that can be produced using the various combinations of OUT(options), used to activate SMF/E, and the Job Card Account Field.

Job Account(n) Field	OUT(option) Chosen at Startup			
Coded Value	**IVP**	**STEP or STEPP**	**JOB or JOBP**	**NONE**
IVP	L,R	L,R	L,R	L,R
STEP or STEPP	L,S	L,S	L,S	L,S
JOB or JOBP	L,J	L,J	L,J	L,J
NONE				
No match	L,S	L,J		

L = JOB LOG Summary Report
J = JOB Detail (SYSMSG)
S = Step and Job Detail
R = Step and Job With Rates Displayed
blank = nothing

The following is a sample procedure that can be used to disable SMF/E:

```
//SMFDEACT PROC X=JARSTATS         SMF/E EXIT ROUTINE TO DISABLE
//SMFE     EXEC PGM=JSYJ1000,
// PARM=('DISABLE(&X.)')
//CAIJRPR  DD  SYSOUT=*
//   PEND
```

As another example, consider the following scenario which shows how IVP can be used by a limited group (for example, systems programmers) to gain familiarity with the product outputs and use of various rates:

1. S SMFE,O=IVP,A=3
    a. The SMFE proc defined on the prior page is invoked here. SMF/E is being initialized in IVP mode.
    b. The 3rd positional parameter is examined for a character string which determines what type of output (if any) is produced.
2. Given that SMFE has been started in IVP mode as in item 1 above:
    a. User address spaces receive output from SMFE *only if* they code a recognizable character as the 3rd positional parameter in the JOB card.
    b. For example: //TEST1 JOB (76251,,IVP),MYNAME,CLASS=A produces the JOB log summary report and step/job detail reporting *with Rate information.*
    c. Another example: //TEST2 JOB (76251,,STEPP),MYNAME,CLASS=A produces the JOB log and abbreviated STEP data. STEPP is suggested for production.

Customization

Customized SMF/E by assembling and link editing a customization table. Specify the load module name of the table to the SMF/E initialization program when SMF/E is enabled. This table controls the SMF/E billing algorithm. A sample table is provided and is used during the installation verification. To build this customized table, use this table as a starter. The source member name is YJ2000 in CAJRSAMP. The JCL for customization is found in CAJRJCL as a member SMFECUST. SMF/E can also be customized using Software Customization Services (SCS) in the CA Mainframe Software Manager. If SCS is used, a copy of the source that is constructed by SCS can be saved for more detailed customization if necessary.

The assembled table is a collection of four or five subtables.

Rate Table

Contains factors and rates for computing the estimated charge for the batch job.

TSO Rate Table

Contains factors and rates for computing the estimated charge for the session.

Priority Table

Contains weighting factors that are applied to the job charge during the charge calculation that are based on the job priority. (The PRTY value that is for the JOB card or default value that is supplied by JES).

Job Class Table

Contains weighting factors that are applied to the job charge during the charge calculation that are based on the job CLASS.

Config Table

Used to categorize devices into one of five groups that are based on the device address/device number. SMF/E can also determine the device type from the UCB type code in the Type 30 SMF record by specifying GNRC(YES) during activation. This method is preferable to coding the CONFIG entries. If using SCS to customize SMF/E, then the Config Table is not available. Use either the GNRC(YES) option during initialization as recommended, or the SCS generated source that can be modified, reassembled, and relinked outside of SCS.

Manual Customization of SMF/E

All five subtables are assembled in a single CSECT and form a single load module. Each subtable is built from three macros:

CAIJS01

Start a subtable

CAIJS02

Enter rates or other variables to a subtable.

CAIJS03

End a subtable

The format of the CAIJS01 macro specification is fixed for each of the five subtables. Code one of the following values at the beginning of each of the five subtables according to the subtable you are generating.

```
CAIJS01  PRIORITY,DEFAULT
CAIJS01  JOBCLASS,DEFAULT
CAIJS01  CONFIG,RANGE
CAIJS01  RATE
CAIJS01  TSORATE
```

Each of the subtables must be terminated with a CAIJS03 macro.

Coding the CAIJS02 macro is more complex than either of the other two. The format varies depending upon which subtable is generated.

1. CAIJS02 statements in the CONFIG subtable have the format:

   CAIJS02 hexlow,hexhigh,device

   Where hexlow and hexhigh are three-digit hexadecimal channel-unit addresses or four-digit hexadecimal device number and provide the boundaries of a range for that device type. The device is the four-character device code that is assigned to each device within the range. Valid device types match the CA JARS CONFIG statement options:

   READ

   Card Reader

   WRIT

   Printer

   PUNC

   Card Punch

   TAPE

   Magnetic Tape Unit

   DISK

   Direct Access Storage Device

   OTHE

   Other

   The CAIJS02 statements order is not important, except when two CAIJS02 statements include the same address, the device is taken from the first CAIJS02 statement.

2. CAIJS02 statements in the RATE and TSORATE subtables have the format:

   ```
   CAIJS02 keyword,factor (or rate)
   ```

   The *keyword* portion of the CAIJS02 statement is a self-defining character string constant and must have one of the values shown in the example. The sample table that is shown here and distributed with the product contains the full set of possible keyword values. Generally, the *factor* value names a rate or factor that you are specifying. All factors can be specified or set to zero if charging on the particular element is not desired.

3. CAIJS02 statements in the PRIORITY and JOBCLASS tables have the format:

   ```
   CAIJS02 value,factor
   ```

Both the *value* and *factor* parameters must be specified. Values in the PRIORITY table must be numeric priority values. Values in the JOBCLASS table must be single character job classes. The factor is a weight which is applied to the job charge. The weight is specified as a percentage where 100 percent means that the weight has no effect. 50 percent would reduce the charge in half, and 200 percent would double the charge. This table is included in CAJRSAMP and can be used as your starter table.

Example with a sample table:

```
YJ2000   TITLE 'JARSTAT RATE TABLES'
YJ2000   CSECT
*                                    JOB PRIORITY TABLE
         CAIJS01 PRIORITY,DEFAULT
         CAIJS02 0,100               PRIORITY 0 OR ANY PRIORITY NOT IN
*                                      THE TABLE BILLS AT 100% X STD
         CAIJS02 13,150              BILL PRIORITY 13 AT 150% X STD
*        .
*        .
*        .
         CAIJS03
*
*                                    JOB CLASS TABLE
         CAIJS01 JOBCLASS,DEFAULT
         CAIJS02 Z,100               BILL JOB CLASS Z, OR ANY CLASS
*                                      NOT IN THE TABLE, AT 100% X STD
         CAIJS02 A,100               BILL JOB CLASS A AT 100% X STD
*        .
*        .
*        .
         CAIJS03
*
*
*                                    RATE TABLE
         CAIJS01 RATE
*
         CAIJS02 BASIC-PROCESSOR-RATE,1000     9999 DOLLARS/HOUR
         CAIJS02 CORE-FACTOR,1.00              99V99 DOLLARS/1K CORE HR
         CAIJS02 ELAPSED-FACTOR,000            999 PERCENTAGE
         CAIJS02 TOTAL-CPU-FACTOR,100          999 PERCENTAGE
         CAIJS02 SRB-CPU-FACTOR,000            999 PERCENTAGE
         CAIJS02 TCB-CPU-FACTOR,000            999 PERCENTAGE
         CAIJS02 READER-I/O-FACTOR,000         9V99 DOLLARS/1000 EXCPS
         CAIJS02 PRINTER-I/O-FACTOR,000        9V99 DOLLARS/1000 EXCPS
         CAIJS02 PUNCH-I/O-FACTOR,000          9V99 DOLLARS/1000 EXCPS
         CAIJS02 TAPE-I/O-FACTOR,5.00          9V99 DOLLARS/1000 EXCPS
         CAIJS02 DISK-I/O-FACTOR,5.00          9V99 DOLLARS/1000 EXCPS
         CAIJS02 OTHER-I/O-FACTOR,000          9V99 DOLLARS/1000 EXCPS
         CAIJS02 CORE-INDICATOR,0              '0' = CORE ALLOCATED
*                                              '1' = CORE USED
```

```
         CAIJS03

*
*                                 TSORATE TABLE
         CAIJS01 TSORATE
*
         CAIJS02 BASIC-PROCESSOR-RATE,1000     9999 DOLLARS/HOUR
         CAIJS02 CORE-FACTOR,1.00              99V99 DOLLARS/1K CORE HR
         CAIJS02 ELAPSED-FACTOR,000            999 PERCENTAGE
         CAIJS02 TOTAL-CPU-FACTOR,100          999 PERCENTAGE
         CAIJS02 SRB-CPU-FACTOR,000            999 PERCENTAGE
         CAIJS02 TCB-CPU-FACTOR,000            999 PERCENTAGE
         CAIJS02 READER-I/O-FACTOR,000         9V99 DOLLARS/1000 EXCPS
         CAIJS02 PRINTER-I/O-FACTOR,000        9V99 DOLLARS/1000 EXCPS
         CAIJS02 PUNCH-I/O-FACTOR,000          9V99 DOLLARS/1000 EXCPS
         CAIJS02 TAPE-I/O-FACTOR,5.00          9V99 DOLLARS/1000 EXCPS
         CAIJS02 DISK-I/O-FACTOR,5.00          9V99 DOLLARS/1000 EXCPS
         CAIJS02 OTHER-I/O-FACTOR,000          9V99 DOLLARS/1000 EXCPS
         CAIJS02 CORE-INDICATOR,1              '0' = CORE ALLOCATED
*                                              '1' = CORE USED
         CAIJS02 TPUT-TGET-RATE,1.00           99V99 DOLLARS/1000
         CAIJS02 CONNECT-TIME-RATE,5.00        99V99 DOLLARS/HR
         CAIJS02 ACTIVE-TIME-FACTOR,100        PERCENTAGE


         CAIJS03
*
CONFIG TABLE                     (illustration purposes only)
         CAIJS01 CONFIG,RANGE
         CAIJS02 150,15B,DISK        DISK DEVICE ADDRESS RANGE
         CAIJS02 180,183,TAPE        TAPE DEVICE ADDRESS RANGE
         CAIJS02 380,383,TAPE        TAPE DEVICE ADDRESS RANGE
         CAIJS02 00E,00E,WRIT        PRINTER DEVICE ADDRESS RANGE
         CAIJS02 00D,00D,PUNC        PUNCH DEVICE ADDRESS RANGE
         CAIJS02 00C,00C,READ        READER DEVICE ADDRESS RANGE
         CAIJS02 450,45F,OTHE        OTHER DEVICE ADDRESS RANGE


         CAIJS03
         END
```

This table can be assembled and link edited into a library that is available to SMF/E activation through STEPLIB or LNKLST.

Customize SMF/E with CA Mainframe Software Manager Software Configuration Services (SCS)

SMF/E can also be customized using SCS along with the Software Deployment Service of CA Mainframe Software Manager.

To verify that the CA JARS Resource Accounting software has been deployed use the Customization Wizard. The panels guide you through this procedure.

Follow these steps:

Define and Target Panel

1. Name your configuration.

   Configurations can be named manually, or a name can be generated.

2. Select the system where this configuration is processed.
3. (optional) Specify a description.
4. Click Next.

Select Functions and Options Panel

To indicate that you want to configure SMF/E, select CA JARS Resource Accounting.

This selection automatically checks the customize SMF/E box.

1. Click Next.

Define System Preferences Panel

This panel changes the settings of system variables. System variables are not used during SMF/E Customization.

1. Click Next.

Create Target Settings Panel

To customization SMF/E, set up the rates and modifiers for SMF/E that are used in constructing the YJ2000 source. The YJ2000 source is used in estimating the costs for batch jobs and TSO sessions.

SYSOUT

Class that is used when the configuration process is implemented.

Default: Asterisk

Any printed output is written to the default output class for started tasks.

Recommended: Change this parameter to a standard held output class at your installation, when the default class is not held automatically.

SMF/E rates

SMF/E rates are preset to the shipped defaults included in the YJ2000 program that is in the sample source library.

To modify and then select the rate to change:

1. Click the set of rates.

Note: The formats of the rates in the default settings are the formats that the SMF/E customization expects.

Example:

> The default Batch DISK-I/O-FACTOR has a value of 5.00 (meaning five currency units per 1000 DASD I/Os). The customization process expects the value to be of the format of *n.nn*. Anything other format can generate an error in customization. The same is true for the TSO Rates. In some cases, the options are sets of switches between a fixed set of values, in this case select one of the values.
>
> **Note:** The default value the switch is set to is the value shipped in the sample YJ2000.

SMF/E modifiers

SMF/E modifiers allow adjustment to charges based on the job class and job priority if desired.

Job classes

All of these modifiers indicate a percentage adjustment.

Default: 100

Format: three-digit number

This 100 percent adjustment indicates that the cost of running a job in that job class adjusts to 100 percent of the cost. This 100 percent means that there is no change to the estimated cost base on the Job Class.

To bill job classes at a cheaper rate, change the percentage to a lower number.

To bill job classes at a premium rate, adjust the percentage upward.

Priority modifiers

Specify up to ten priority modifiers.

Default: First entry is the default percentage for all unspecified priorities.

Format: *pp,nnn* where *pp* is the two-digit priority and *nnn* is the three-digit percentage.

Recommended: Set the first entry to *00,100* meaning that the default priority adjustment is 100 percent.

To set verify the priority modifiers default:

1. Reply *yes* to the priority modifier question by acknowledging that you noticed that this priority modifier is the default.

   Note: When more than ten priority adjustments are needed, enter the default and nine other priority settings. Once configuration is complete and the source is generated, edit the source directly outside of SCS to add the additional priority subtable entries.

Assembly work unit

This option is used in the source generation, assembly, and the linking process.

Specify a DASD class for the SCS file allocation.

Default: SYSDA

Note: These files are not temporary files as traditionally defined, in that they are allocated permanently during the configuration process. These files are deleted once configuration completes successfully.

Save the generated product.

This option saves the generated product source.

Default: Yes

Recommended: Save the source for future reference.

To save the source produce, by customization:

1. Answer *yes*.
2. Specify the dataset name where the generated source will be written.

   Note: This specified dataset must be preallocated with an LRECL of 80 bytes and a RECFM of FB or F.
3. Enter the dataset name of the generated SYSLIN dataset from the assembly that is used to create the customization load module.

   Note: The SYSLIN dataset is not a temporary dataset in the traditional sense, but it is allocated and deleted during the implementation of this configuration.
4. Click Next.

Select and Edit Resources Panel

No changes for this panel.

1. Click Next.

Review and Build Panel

To review the options and setting for SMF/E:

1. Click *Target Settings.*
2. If the settings are valid, click *Build*.
3. To make any needed changes, click *Back*, until you are at the Target Settings.
4. Make any necessary adjustments, click *Next*, until are back on the Review and Build panel.

Configuration built

When the configuration is built, implement this configuration in SCS.

The output load module with the configuration information (JSYJ2000) is linked edited into the deployed CAJRLOAD library. The generated source is saved into the named specified dataset (when save dataset was selected).

Chargeback Algorithms Used

SMF/E does not use all of the chargeback features available to CA JARS. However, this is not a limitation as the subset supported is that portion of the chargeback algorithms most widely implemented by current customers. The discussion that follows starts with a final figure and proceeds to finer levels of detail as the ingredients of the final figure are explained.

BATCH Charges

1. The Total Charge is calculated as:

   ```
   Total Charge in Dollars = (Sum of the step charges x
                             Priority Factor x
                             Job Class Factor)
   Step Charge in Dollars =  (Processor Charge + I/O Charge)
   ```

   The only adjustments applied at the job level are Priority and Job Class factoring. The Priority and Job Class tables contain these factors.

2. The Processor Charge is computed on a per step basis as follows:

   Processor Charge = Processor Time x Adjusted Rate

3. The time factor in the Rate Table enables a user to weigh or delete the various recorded times when calculating the processor time.

   The Processor Time is calculated as follows:

   ```
   Processor Time = ((Elapsed Time   x Elapsed-Time-Factor) +
                     (Total CPU Time x Total-CPU-Factor)    +
                     (SRB CPU Time   x SRB-CPU-Factor)      +
                     (TCB CPU Time   x TCB-CPU-Factor))
   ```

   This algorithm permits a charge which includes all four times at 100%. Obviously, this results in overcharging. Though the algorithm permits this overcharging, the recommended approach is to charge for Total CPU alone at 100%. The other factors should be set to zero. Total CPU is the most repeatable and uniform measure of CPU utilization for a given processor and, therefore, is likely to be the most equitable variable to use.

4. The Adjusted Rate allows for charging to be a function of memory utilization as well as CPU time. The Core Factor in the Rate Table lets you charge for the utilization of memory as a function of time (duration of use). This dollar amount is charged per 1K block of core. The Core Indicator (in the Rate Table) selects either Core Used or Core Allocated. In the formula below, this choice of memory measured is shown simply as the word *CORE:*

   ```
   Adjusted Rate = Basic-Processor-Rate +
                   (Core x Core-Factor)
   ```

5. The I/O Charge is calculated by applying the appropriate factor to EXCP counts (if available) for each device, and adding the individual costs. If a device falls in an address range specified in the CONFIG Table or if the GNRC(YES) option is used, then the corresponding device factor from the Rate table is applied.

The I/O Charge is calculated as follows:

```
I/O Charge = ((Reader  I/O Count x Reader-I/O-factor)  +
              (Printer I/O Count x Printer-I/O-factor) +
              (Punch I/O Count   x Punch-I/O-factor)   +
              (Tape I/O Count    x Tape-I/O-factor)    +
              (Disk I/O Count    x Disk-I/O-factor)    +
              (Other I/O Count   x Other-I/O-factor))  / 1000
```

The result of the computations is an estimated step charge which is summed to produce the total estimated job charge. The charge is estimated and does not balance exactly to a CA JARS batch run due to a number of factors. Rounding is one factor. Another factor is that SYSOUT charges are not included in this estimated charge. SYSOUT charges are based on the SMF Type 6 record which is usually written sometime after the job terminates and is therefore unavailable for cost calculation in SMF/E.

TSO Charges

1. The Total Charge is calculated as:

```
Total Charge = Processor Charge + I/O Charge
               Connect charge +
               UR Charge
```

2. The Processor Charge is computed as follows (same method as for batch):

```
Processor Charge = Processor Time x
                   Adjusted Rate
```

3. Processor Time is calculated as follows:

```
Processor Time = (Active Time x
                 Active Time Factor) +
                 (Total CPU Time x
                 Total CPU Time Factor)
```

4. The Adjusted rate is computed the same as described earlier for batch.

5. The I/O Charge is calculated the same as described earlier for batch.

6. Connect Charge for TSO sessions is computed as follows:

```
Connect Charge = Connect Time x
                 Connect Time Factor
```

7. The Unit Record Charge allows for chargeback against terminal activity. It is computed as follows:

```
UR Charge = ((TPUTS x TPUT-TGET-Rate) +
             (TGETS + TPUT-TGET-Rate)) / 1000
```

System Requirements

The following system related facts should be considered when running SMF/E:

- SMF/E utilizes CAIRIM, one of the CA Common Services for z/OS CAIRIM is a collection of dynamic initialization routines that eliminate the need for User SVCs, SMF exits, subsystems, and other installation requirements required in prior releases of this product.
- CAIRIM must be run with the specific CA JARS family PARM prior to the activation of SMF/E. Running CAIRIM does **not** activate SMF/E. See the *Installation Guide* for more information.
- SMF/E no longer uses SMF exits. All SMF information used by SMF/E is obtained by a dynamic intercept that gains control prior to any SMF exits. The intercept is done using dynamic front ends, where we branch to our code and then back to the real IBM routines. No changes to the operating system, no replacements of IBM modules, and no overlays of IBM code are used. The technology is as reliable as the SMF exit conventions themselves.
- SMF/E requires that the target library, CAI.CAJRLOAD, be in the Linklist and APF-authorized.
- A small amount of CSA, probably under 4K, is needed by SMF/E.
- SMF/E messages CAJR250I, CAJR251I, and CAJR252I are written to the JOB LOG for each job. These messages are written with a Write-To-Operator (WTO) specifying ROUTCDE=11. For those jobs (notably started tasks) that have no RPL for the Job Log data set, these SMF/E messages appear in the SYSLOG. Any consoles in the complex servicing ROUTCDE=11 receives these messages. If you do not want them to display on any particular console, then change the console ROUTCDE so that ROUTCDE=11 is not serviced. Alternatively, the messages can be suppressed entirely from all consoles in your complex by specifying message suppression in MPFLSTxx (Message Processing Facility List) for the three messages.
- Once SMF/E is customized to your installation requirements, a procedure should be installed in a proclib that can be used to activate and deactivate SMF/E. The activation procedure can also be started through your COMMNDxx SYS1.PARMLIB member. This ensures that SMF/E is started after every IPL.

Chapter 3: Performance Considerations

The Job Accounting Report System provides you with a number of options for managing and processing job accounting data. The *summarization* (or history file) feature is one of the options. The advantage of this feature is that once the basic input has been processed, a condensed file can be created and saved for future reporting. The summarization (or history file) feature enables the creation of a history file simultaneously with the production of a user-defined report. Up to 15 history files (one for each report) may be produced during one execution of CA JARS. These history files can serve to reduce the file to only those levels needed to meet specific reporting requirements. This feature can save considerable time in processing job accounting data and information.

The history file feature is designed to offer many benefits for user reporting:

- For many reports, a greatly reduced number of records is processed saving significant machine time.
- Since summary records are typically already sequenced for a given report, sort time is dramatically reduced.
- Summary records can be reprocessed to create yet smaller files.
- Processing time for *one-time-only* reports becomes insignificant.
- Use of the most detailed option of the history feature allows the retention of all detail contained in the raw data including any EXTDATA records.

History files can be created in each user-defined report by specifying history output on the SORT statement.

History files are divided into two categories:

- Detail: one record created for each job (level 6), or one record for each job and for each job step (level 7). When detail is specified, EXTDATA records are placed in the output file.
- Summary: one record created for the most detailed sort level defined on the SORT statement and requested as the history level (levels 1-5). At summary level, EXTDATA records are not retained.

This section contains the following topics:

History Records Compared to Account Records

The history records for the detail files are the basic account record format (this record is displayed in the section under the account exit) *with the following adjustments:*

1. File level (field position 2) reflects the ID of the record type this record was created from. See the *User Guide*.
2. Group codes are present if applied using the grouping feature in the report which created this file.
3. CPU ID reflects any changes made through report grouping.
4. Devices used by this step or job are present in the I/O device table which also contains device type and count.
5. Sysout table is present in the job records.
6. If rates have been applied, they are carried in this record.
7. Records rejected by grouping or the DEVADDR or DEVNMBR statement are not on the file.
8. Any changes made to a record in an input, account, exit1 or exit2 exit (if used in this report) are carried in the history file.
9. System records are also written to these history files. The record description for these records is on the next page.

```
01 SYSTEM-LEVEL-ACCT-RECORD.

   02 XA-RDW.
      03 XA-RDW-LRECL                         PIC S9(4) COMP.
      03 XA-RDW-ZEROS                         PIC S9(4) COMP.

   02 XA-CPU-ID                               PIC X.

   02 XA-FILE-ID                              PIC XX.

   02 XA-RECORD-STAMP.
      03 XA-DATE (YYMMDD)                     PIC X(6).
         (FOR IPL RECORD, DATE OF IPL)
         (FOR SMF DATA LOST RECORD, DATE OF START OF LOST PERIOD)
         (FOR WAIT TIME RECORD, DATE OF START OF TIME PERIOD)

      03 XA-TIME                              PIC X(6).
         (FOR IPL RECORD, TIME OF IPL)
         (FOR SMF DATA LOST RECORD, TIME OF START OF LOST PERIOD)
         (FOR WAIT TIME RECORD, TIME OF START OF TIME PERIOD)

   02 XA-JOBNAME.
      03 XA-JOBNAME-FILLER                    PIC X(4).
      03 XA-SID                               PIC X(4).

   02 XA-FILLER                               PIC X(4).

   02 XA-PROCESSING-ID                        PIC X.
      (@ =IPL, ? =SMF DATA LOST, # =WAIT TIME)

   02 XA-ELAPSED-TIME                         PIC S9(6)V9(5) COMP-3.
      (FOR IPL RECORD, LENGTH OF IPL-IDLE TIME)
      (FOR SMF DATA LOST RECORD, LENGTH OF LOST PERIOD)
      (FOR WAIT TIME RECORD, LENGTH OF INTERVAL)

   02 XA-DATA-VALUE                           PIC S9(6)V9(5) COMP-3.
      (FOR IPL RECORD, 0)
      (FOR SMF DATA LOST RECORD, 0)
      (FOR WAIT TIME RECORD, LENGTH OF INTERVAL)

   02 XA-LEGIT-RCD-FLAG                       PIC X.
      (FOR WAIT TIME RECORD, FLAG = 1 INDICATES THE RECORD IS FOR
      AN ACTUAL IPL AND NOT A SPLIT RECORD DUE TO SPANNING MIDNIGHT.)

   02 XA-PROCESSOR-WAIT-TABLE.
      03 XA-WAIT-TIMES       OCCURS 16 TIMES  PIC S9(6)V9(5) COMP-3.

   02 XA-ONLINE-FLAGS.       (A 1 INDICATES PROCESSOR IS ONLINE.)
      03 XA-PROCESSOR-ONLINE OCCURS 16 TIMES  PIC X.
```

Summary File Examples (Levels 1-5)

Summarized history files are created after the sort and prior to the creation of the output data elements table.

Again, the records are in the basic account record format. A level-5 file contains the most records and level-1 the least.

1. File level (field position 2) is a 1-5 and be the same as the history-level flag on the SORT statement.
2. Processing ID (field position 28) is an H.
3. Sysout table is absent. Total standard lines, total special lines, total lines printed and pages printed are retained in the record.
4. The I/O device table is not created. Total I/O counts by device type and total counts of the individual device types are present in the record.
5. All fields which are *additive* or capable of being mathematically accumulated are present and reflect the totals from the records in that sort break.
6. Since character fields are not cumulative, they are usually set to blanks, **unless** the field was one of the sort keys at the time the file was created.
7. EXTDATA is absent.

Example:

1st sort level is job start date (month and year only).

2nd sort level is a 4 position accounting field that reflects departments.

3rd sort level is job name.

A level-3 history file contains one record for every unique job name run for an individual department for specific month and year.

If job A has been run 4 times for the accounting department in March of 1988, then one record is written to the history file containing all the accumulated statistics for four executions of job A. It also contains the accounting information and month/year values.

A level-2 history file, based on the above information, has one written for each department for a specific month and year. The accumulated statistics are for *all* jobs run for that department, in that month and year. Therefore, in contrast to the level-3 history file, a level of detail is missing because the file does not have statistics for each unique job name within the department.

Considerations When Using a Summarized History File

The differences between summary history file records and records contained in the Basic Accounting Table (or detail history records, level 6 and 7) give rise to these usage considerations:

- Data elements used in sort control field definitions are filled in only to the level specified by the history level on the SORT statement.
- All data elements which can logically be summarized are included in the history record regardless of whether they are specified in the DISPLAY statement for the report. If in doubt, use the Output Data Elements Table in the *User Guide*. Any data element which can appear in a summary line is in the summarized history file.
- Data elements in the history record which pertain to computer billing are not filled with valid data unless rate parameters are present during the creation of the summary file. Rate parameters are input through the use of the RATE, TSORATE, FORMRATE and RJERATE statements.
- Summarized history files can be processed by CA JARS to produce user-defined reports and additional history files, subject to the following limitations:
  - Reports designed to utilize the history file as input cannot provide data at a level of detail greater than that specified at the time the summarized file was initially produced.
  - CA JARS does not accept summarized file input with mixed record types (i.e., job records cannot be processed simultaneously with records produced at a sort break). (You cannot run a level-2 history file with a level-7 history file.)
  - When summary file input is present, RATE statement parameters cannot be applied or used to override existing data elements unless the summary file was created at the step level (level 7).
  - Distributed charge displays only if the file was created in a report which used a RATE statement.
  - Credit and Debit amounts appears as part of total charge.
- System Use Reports can be requested using detail history files created from raw SMF data.
- Device type and addresses can not be overridden in subsequent runs by using a history file as input.

Creating a History (Output) File

The presence of a history file ddname in the SORT statement indicates to CA JARS that a history file is to be produced (positions 55-62). The history-level flag (position 63) must be coded to define the level of detail to be contained within the history file.

The ddname requested on the SORT statement must also have an output JCL statement in the run deck. For example, if the SORT statement contains the ddname HISTORY3, the JCL statement would be:

```
//HISTORY3 DD DSN=...
```

Using a History (Input) File

The history file can be subsequently introduced using the History Input Indicator. The ddname for an input history file is CAIJSHST.

Note: When processing history input with SMF data, duplicate record checking occurs only for SMF data.

Grouping

Grouping is one of the most powerful features of CA JARS. It can be used to:

- Select/reject account records from processing in a given report
- Group together account records based on an indicated criteria
  - Group codes can be assigned for sorting and/or grouping
  - CPUID can be altered to sort or group records so multiple billing algorithms can be applied

Grouping Considerations

- Grouping is applied to step and job records--including 1 step job records created by CREDIT, DEBIT and BUDGET statements
- Grouping occurs prior to rate application
- Limitations
  - One GROUP statement only: Up to six Basic Account fields defined
  - Unlimited GROUPC statements per report
- The GROUP statement along with the GROUPC statements define the data to be selected or rejected based on the actual values found in specific positions in the Basic Accounting Table
- AND Logic

  Grouping is an AND type of logic. Qualification starts with the first test. If selected, the account record then goes to the next test (if any) and so on through all fields to be tested. If a record meets any test, it is rejected or selected based on the rejection/selection indicator.

- GROUP statement
  - Limited to 6 tests (applied in sequence)
  - States the offset in the account record to check
  - Number of positions of data to look at (up to 8 positions)
  - Format data is carried in (hexadecimal, packed or alphanumeric)
- GROUPC statement
  - Must be associated with GROUP statement
  - Sets ranges of test (high value/low value)
  - Unlimited number of GROUPC statements permitted
  - Ability to reassign or assign CPUIDs and/or group codes
- OR Logic

  Grouping can also be an OR logic by using the grouping-on-grouping technique. By not selecting or rejecting the account records, test for a value or range of values and, based on these values, assign unique group codes. These group codes can then be tested and the account records which qualify can then be selected or rejected.

Retaining the Intermediate Account File

CA JARS automatically creates an intermediate working data file called the *account* file. This file is a concise and manageable collection of computer utilization statistics at the job and step levels and can be defined as a temporary sequential file.

You can elect to keep this file for future processing by defining a permanent data set for ddname CAIJSACT in the JCL.

Suggested reasons for keeping this intermediate file are listed below, along with a discussion of some appropriate SELECT flags.

1. The No Input Indicator

   If you intend to produce more than 15 user-defined reports with the same data, the account file may be retained using the *no input indicator,* (position 16 on the SELECT statement). A second step in the execution job stream can produce up to 15 more reports. No new input is allowed in the second step. Performance for this step is enhanced over that of the first step, in that the account file is not reconstructed. Processing begins in Phase 4. Refer to the chapter titled "System Description" in this guide.

2. Account Input Indicator

   The retained account file can be introduced to CA JARS in a later job stream using the *account input indicator.* Refer to the SELECT statement. Any user-defined report normally produced using the *raw* SMF or DOS job accounting data, can be generated using the account file as input at a considerable savings in processing time.

   It should be noted here that the account file record layout is subject to change from one release to another of this product. Therefore, this file should not be used for historical backup or year-to-date reporting purposes. Use history output for those purposes. The sole function of the account file is to relieve the user of the need to repeatedly enter the *raw* SMF or DOS job accounting data to generate multiple reports against a given amount of data. Saving the account file should not serve as an alternative to creating a history file. Reprocessing a history file is the preferred method of processing except in the case where more than 15 user-defined reports are to be generated against the same account file.

   Note: In this case, the account file contains records corresponding to all available input. If you were to input historical data through the ddname CAIJSHST, the output account file would also contain these history records in addition to account records. When such summary records are present, the account file should not be retained and reused.

3. EXTDATA Records

   If *EXTDATA records* are to be the primary output of JSIMAIN, then for efficiency, it makes sense to make the Account file permanent, and not to produce a history/summary file.

Chapter 4: Special Usage Considerations

The Distributed Charge feature allows you to implement a cost recovery technique of chargeback. The Operating Cost, specified on the PARMS statement, is proportionately distributed to all jobs based on the percent of resources each job consumed.

The Distributed Charge feature is an immediate way of showing how much of the total operating cost is attributable to an individual job or a department.

A RATE statement is required so the report program calculates a total charge and a percent of total charge for each job. It is the percent of total charge that determines what percent of the operating cost is attributed to each individual job.

```
Job Distributed Charge =

Operating Cost * Percentage of Total Charge
________________________________________________

              100
```

Distributed Charge and Total Charge are mutually exclusive.

Because of rounding in the calculation of the percentage of total charge and in the calculation of charges, it is possible to recover 98%, 99% or even 101% or 102% of the operating cost specified on the PARMS statement. This is caused by rounding and is to be expected.

Note: JARS/OLF supports its own method of Cost Allocation and Recovery. If JARS/OLF is to be used for chargeback, then see the *JARS/OLF User Guide.*

This section contains the following topics:

Account Code Conventions

CA JARS provides a flexibility that lets you define your own accounting convention for reporting purposes. The standard method of allocating computer usage back to various users starts with the creation of accounting or project codes within an organization. These codes often take on varied meanings such as:

- account and subaccount numbers
- employee identification
- application codes (payroll, accounts payable, etc.)
- type of run (production, test, rerun, etc.)
- program class (compile, link-edit, tape-copy, etc.)

CA JARS assumes you have implemented the necessary SMF or JES exits that validate accounting data.

An added concern for those installations processing under both DOS/VSE and z/OS is to ensure that the job accounting conventions set up for the two systems coincide. Reports reflecting total utilization of a computer under two systems are more meaningful if the data are under the same conventions (whether it be codes in the Jobname or User Information fields).

Account Codes in z/OS JOB Statements

This product can retain a maximum of 48 bytes of accounting data from raw SMF records.

The following illustrates how a typical JOB accounting string is processed.

```
//MYJOB JOB (1234,ABCD,X,Y,Z),'PROGRAMMER.NAME',
 //          MSGLEVEL=1
```

The Accounting Field information, without the intermixed commas (,), appear in the corresponding field of the job and step records as follows:

```
1234ABCDXYZ
```

If less than 16 bytes of accounting data is coded (as in this example), the character string is left-justified and placed in positions 77-92 of the account record. More that 16 bytes causes an overflow into positions 487-502, as needed, and is always left-justified.

If a positional subparameter of the accounting information is omitted, the passed data is arranged left to right. That is, in the above example if the data were (1234,,X,Z) the field would become 1234XZ.

Account Codes in z/OS EXEC Statements

The ACCT parameter on the EXEC statement can be used by installations that want to perform computer billing or cost distribution at the job step level. An example of this would be to bill one account for the execution of the first job step and a different account for the second job step of a job, and so forth. When designing a report of this type, work only with step records. This can be accomplished by placing an S in the Required Records field on the SORT statement for that report. Otherwise, job records are included in the report and duplicate billing occurs for the same job step.

You should use discretion when allocating charges at the step rather than the job level. Step level charging precludes your ability to charge for U/R, special forms, and job setup (tape and disk mounts) since these charges are only available at the job level. Therefore, step charging should be avoided. When accounting information is absent from an EXEC statement, CA JARS assumes that the accounting information on the JOB statement is to be used. In other words, the Accounting Field information in a job record is propagated into each of the subsequent step records having a blank Accounting Field.

Account Codes in VSE

The VSE job accounting records as generated by the CA $JOBACCT Interface Routine contain fields similar to those in the SMF records. These fields can be employed in the same manner for accounting purposes.

The User Information on the VSE JOB statement is carried through CA JARS on each job step record for the job. The User Information captured by the Job Accounting Option in the DOS supervisor operates in the following manner:

```
// JOB MYJOB 1234ABCDXYZ
```

The User Information field begins with the second position following the last character of the Jobname. This means that the field *floats* on the JOB statement, depending on the length on the Jobname. A maximum of 16 characters are picked up and placed in the User Information field without any alteration.

SMF Job Accounting

This section applies only to JARS Account record processing, not to EXTDATA processing. The following SMF record types, if present, are processed to generate user-defined job accounting reports:

Type	Description
6	Output Writer

Type	Description
26	Job Purge
30	All subtypes
33	APPC
47/52	RJE Session Log-On
48/53	RJE Session Log-Off
49/54	RJE Session Invalid Log-On attempt

The basic unit of work for batch jobs and TSO sessions is the step, represented by the Step Termination or TSO-Step Termination SMF record.

If at least one of these types of records or at least one type 6 record (Output Writer) is present, a job or one-step job account record is created. Utilization data is not taken from a type 30-5 record (Job Termination) in the absence of type 30-4 record.

The job termination record and logoff (30-5) records are written by SMF at job termination and logoff times, respectively. Information taken from these records includes the Programmer Name and Job Accounting fields (CA JARS accumulates most other job-level data from the step level).

For various reasons, including machine failure while a job is processing, the 30-5 record for a job may be missing. In the absence of a 30-5 record, the Programmer Name and Job Accounting fields are taken from the Job Commencement (30-1) record, if one is available.

Step account records are constructed from the accumulation of data in subtype 2 and 3 records. The SMF interval start date and time from these records is used as the step Start Time, allowing jobs that continue to execute past an accounting period to be reported on properly within each period.

The Job Commencement (30-1) record is only used to supply the Programmer Name and Job Accounting fields for jobs that lack 30-5 records.

Output Writer (type 6) SMF records are written at the completion of each Unit of SYSOUT processing (printing or punching) for a job. Information from these records is used to build SYSOUT Table entries which are appended to the job account record.

Note that if at least one type 6 SMF record is present for a job, then a job account record is produced. If no step-level records are present, the job has zero steps and zeros for all utilization statistics. If no type 30-5, 30-1, or 26 is present, then the accounting information fields are blank.

The Job Purge (type 26) record provides the following information: the Job Class, Priority, Job Number, Input Device Name, Account Code at appropriate JES release, the Input Route code (for RJE users--JES2 only), and the number of statements read by the job.

In the absence of a type 26 record, the Job Class and Priority are taken from the type 30-5 record, if one is available. The number of statements read is accumulated from the type 30 records for the job (in which case JCL statements are not included in the total).

Other SMF records processed by CA JARS for the purpose of gathering system-wide utilization statistics are as follows:

Type	Description	Use
0	IPL	IPL Idle Time
7	Data Lost	SMF Data Lost Time
70	CPU Activity (z/OS only)	Wait Time

EXTDATA SMF Job Accounting

EXTDATA records are basically a reformatting of the input SMF records. CA JARS applies certain tests to ensure that these records contain appropriate Account codes for later reporting and Chargeback.

Type 30 records are processed according to their subtype to create the appropriate EXTDATA records.

30-2 or 30-3	->	EXT30 Interval and EXT30 IO Interval (S30I, SIOI, U30I)
30-4	->	EXT30 Step and EXT30 IO Step (S30X, SIOX, U30X)
30-5	->	EXT30 Job and EXT30 IO Job (S30, SIO, U30)

Each type 6 record causes an EXT06 record to be produced. EXTDATA records can only be produced by turning on the appropriate option in the EXTDATA statement.

EXTDATA processes may alter SMF records. See the SMF Record Usage table in the "System Description" chapter for more information.

SMF Audit Flags

The output data element F9 is provided as a debugging tool so you can report which critical SMF record types have been used by CA JARS to create the job or step level account record. This data element is not available at the summary level. It is only valid for detail (job or step level) history records.

When F9 is displayed on a report, each detail line shows a string of six characters (e.g., yyynnn). Each position represents a particular SMF record type as stated below. **Y** means record type was used; **N** means it was not used.

Position	SMF Record Type
1	30-5
2	6
3	26
4	47, 53
5	30
6	30-2/3

Interpreting SNAPDUMP Output

The CA JARS Report Writer performs extensive validity checking on the SMF input file. Blocks of SMF data are read and deblocked by the Report Writer, rather than the operating system, in order to give the program greater control over the action to be taken when errors are encountered. SMF records are checked for invalid data as they are read. The following are the reasons for this validity checking:

1. To allow processing to continue, if possible, in spite of the presence of invalid data in the input data set.

2. To prevent the inclusion of erroneous data in the output Job Accounting reports and history files.

When invalid data is encountered, it is discarded. Message CAJS941I at the end of the input phase of a run indicates that invalid data has been detected. This message gives the number of segments and the number of blocks discarded because of I/O errors or errors attempting to deblock record segments with invalid control information in the segment description words. Message CAJS944I gives the number of additional records that have been discarded for failure to conform to correct SMF record formats.

When data is discarded, it is also displayed on the SNAPDUMP output file by means of the OS SNAP macro. A maximum of ten SNAPs are issued by the Report Writer in any given run. (Thereafter, discarded data is not SNAP'd, but the discarded blocks, segments, and records continue to be counted.)

The first item in each SNAP is one of the following Report Writer control blocks: the GETCB, which contains information describing the status of input and deblocking operations against the SMF input file; or the RCAREA, which contains information describing the current status of SMF record processing. The contents of these two control blocks is described below.

Following the GETCB or RCAREA, the data in error is displayed. (The exact data displayed depends on the reason for the SNAP, as described below.)

At the end of each SNAP, a message appears containing a one-letter reason code which is used to determine the cause of the SNAP. The following paragraphs explain possible reason codes.

SNAP Reason Codes

The following reason codes describe errors that occurred while reading or extracting segments from a block of SMF data:

A

I/O error on CAIJSSMF. Message CAJS951I gives information on the nature of the error. The GETCB is SNAP'd to help locate the exact block where the error occurred. Processing resumes with the next block. (However, processing terminates after three consecutive I/O errors.)

B

The block descriptor word at the beginning of a block of SMF data contains a length value less than 9. The GETCB and the block are SNAP'd. Processing resumes with the next block.

C

The segment descriptor word at the beginning of a block of SMF data contains a length value less than 4. The GETCB and the block are SNAP'd. Processing resumes with the next block.

D

The segment descriptor word of the segment being extracted from a block of SMF data contains a length value indicating that the segment extended beyond the end of the block. (The probable cause is a truncated block.) The GETCB and the block are SNAP'd. Processing resumes with the next block.

The following reason codes describe errors that occurred while attempting to assemble segments into a complete logical record in the program work area:

E

The span control bits in the first segment to be moved to the work area indicated that it was not the first segment of a record. The GETCB and the segment are SNAP'd. Processing resumes with the next segment.

F

The span control bits in the current segment indicate that it is the first or only segment of a record; however, no final segment has yet been encountered for the partial record currently being assembled in the program work area. The GETCB, the segment, and the contents of the work area are all SNAP'd. Processing resumes with the current segment.

G

The current segment cannot be assembled into the program work area because the work area is too small. (See note 2 in the section Notes on Common Problems in this chapter for a discussion of how to control the size of the work area.) Processing resumes with the next segment.

H

A discontinuity has occurred on the input file before an end segment has been encountered for the logical record currently being assembled in the work area. The most probable cause of the discontinuity is that a block has been SNAP'd for reasons A through D above.

A discontinuity can also occur during data set switching for concatenated input if the LRECL value for the new data set is larger than the existing work area size. (When this happens, the work area must be freed and reallocated. The GETCB is SNAP'd, along with the contents of the work area, if possible. Processing resumes with the next segment.

I

End-of-file has been reached before an end segment has been encountered for the logical record currently being assembled in the work area. The GETCB and the contents of the work area are SNAP'd. Processing continues as if normal end-of-file had been reached.

The following reason codes indicate that a data exception or other program check has occurred in the course of processing an SMF record, or that invalid or inconsistent data has been detected in the record. In each case, the RCAREA and the SMF record in question are SNAP'd and processing resumes with the next logical record.

If a user SMF input exit is being used to reconstruct SMF records in a work area, both the original record and the reconstructed record are SNAP'd.

The SNAP'd SMF record(s) should be checked against the record descriptions in the IBM *SMF Reference Manual.*

J

A program check has occurred during mainline CA JARS processing. This should never occur. If it does, contact CA Technical Support for assistance. For online technical assistance and a complete list of locations, primary service hours, and telephone numbers, contact Technical Support at http://ca.com/support.

K

A program check has occurred while validating the information in the current record. Probable cause is an invalid Julian date in the record.

L

A program check has occurred while extracting information from the current record. Probable cause is an invalid Julian date in the record.

M

A program check has occurred while the user SMF input exit had control. Interrupt information stored in the RCAREA can be of use in locating the cause of the problem.

N

Invalid data has been detected while validating an SMF record. Possible causes include: a record length shorter than the allowable minimum for that record type; inconsistencies in the internal length fields that define subsections of certain record types; or, for SMF type 6, SMF6SBS is X'0000'.

O

Invalid data has been detected while extracting data from an SMF record. The probable cause is an inconsistency in the internal length fields of the record.

P

An error has occurred while attempting to normalize a time/date stamp in a record relative to the Reader Start Date for the job. The difference between the two dates is greater than 50 days, either positive or negative, indicating probable erroneous data.

Q

The SMF buffer date on a record has exceeded the expiration date of your version of the CA Job Accounting Report System. If the problem continues to occur, contact CA Technical Support. For online technical assistance and a complete list of locations, primary service hours, and telephone numbers, contact Technical Support at http://ca.com/support.

R

The CMF record does not match any of the Data Dictionaries provided.

1

Record too short. Record is SNAP'd and rejected.

2

Unknown subtype in SMF30 header.

3

Unknown subtype in SMF30 subsystem section.

4

Invalid subtype in SMF30 subsystem section.

5

SMF30-1 record did not contain an ID section.

6

SMF30 record did not contain required sections.

7

SMF30-5 record did not contain an ID section.

8

Subsystem section missing from SMF30 record.

Notes on Common Problems

1. If you have a large number of SNAPs with a reason code of D, you should review the processing sequence that created your SMF input data set to determine at what point the blocks in the data set might have been truncated.

   Another possibility is that incomplete blocks are being read by the Report Writer. This happens if the program receives an incorrect blocksize specification for the CAIJSSMF file. Check that the blocksize specified on the CAIJSSMF DD statement, if any, is at least as large as the actual blocksize of the file. If a blocksize has not been specified, verify that the blocksize in the data set label is correct. If in doubt, increase the blocksize on the CAIJSSMF DD statement and rerun the job.

2. If you have a large number of SNAPs with a reason code of G, the Report Writer is probably not allocating a large enough work area to hold the largest SMF records in your input file. The size of the work area is determined by the LRECL value of the input data set, obtained either from the LRECL subparameter on the CAIJSSMF DD statement, if any, or from the data set label.

   In addition, if the RECFM attribute of the CAIJSSMF data set is VB, specified on the DD statement or on the data set label, then the program does not allocate any work area at all for spanned record assembly. (The work area size being used by the program can be determined by examining the GETCB in the SNAP.)

If necessary, change the RECFM subparameter on the CAIJSSMF DD statement to RECFM=VBS, or increase the record length specified by the LRECL subparameter and rerun the job.

The GETCB Control Block

The following control block is used by CA JARS when reading SMF records (Phase 2):

GETCB Control Block Layout

Offset	Format	Content
0	CL8	'GETCB' - Identifies this control block.
8	CL8	(internal use)
10	F	Length of work area for assembling spanned records.
14	A	Address of work area.
18	A	Address of current block of SMF data.
1C	A	Address of end of current block.
20	A	Address of current record segment within the block.
24	A	Address of end of current record segment.
28	H	Length of current block.
2A	H	Length of current segment.
2C	H	Length of record in work area, if any.
2E	CL4	(internal use)
32	PL2	Current data set number within concatenation.
34	PL4	Current block number within data set.
38	PL4	Number of records processed in locate mode in the buffer.
3C	PL4	Number of records that had to be assembled in the work area from spanned record segments.
40	PL4	Number of blocks of SMF data discarded.
44	PL4	Number of times spanned record assembly could not be completed.
48	PL4	Block number where first I/O error occurred in a series of I/O errors.
4C	PL2	Number of consecutive I/O errors that have occurred.

Offset	Format	Content
4E	PL2	Consecutive I/O error limit (constant).
50	CL6	CAIJSSMF Data Control Block (DCB).

RCAREA Control Block

The following control block is used by CA JARS when formatting SMF transmittal records (phase 2):

RCAREA Control Block Layout

Offset	Format	Content
0	CL8	'RCAREA' - Identifies this control block.
8	A	Reserved
C	A	Address of last SMF record read from CAIJSSMF.
10	CL2	(not used)
12	H	Length of last SMF record read. **Note:** The contents of the next three fields are identical to the contents of the previous three fields, except when a user input exit is changing or inserting records.
14	A	Address of SMF record currently being processed.
18	H	Length of SMF record currently being processed.
1A	H	(not used)
1C	CL12	(internal use)
28	PL4	Number of records deleted by user input exit.
2C	PL4	Number of records inserted by user input exit.
30	PL4	Number of records that failed date criteria checking.
34	PL4	Number of records that passed date criteria checking.
38	CL8	(internal use)
40	PL4	Number of invalid SMF records.
41	PL4	Reserved **Note:** The following fields are filled in for SNAPs that were caused by a data exception during SMF record processing (reason codes J, K, L, and M).

Offset	Format	Content
48	CL8	Program status word (PSW) at the time of interrupt.
50	CL8	Contents of general registers 14 - 15 at the time of interrupt.
58	CL56	Contents of general registers 0 - 13 at the time of interrupt.

VSE Job Accounting

VSE History Files created by CA JARS/VSE can be input to this product for processing through the JSITOPS Utility Program. See the section DOS History File Conversion Utility in the chapter "Utilities" for details.

Integration with CA Service Desk

In the unlikely event of an abnormal termination, CA JARS can open requests in CA Service Desk. To use this feature, you must do all of the following:

- Install CA Service Desk on an appropriate platform. See the CA Service Desk documentation for supported operating platforms and other information about CA Service Desk.
- Install and configure the CA Service Desk interface on a mainframe system. In addition, the CA Common Services SOAP client must be installed and active on the system where CA JARS is running or on a system reachable via CCI. For more information on the SOAP client, see the CA Common Services documentation and visit http://www.w3.org/2000/xp/Group.
- Configure CCI to allow for communication with the SOAP client if CA JARS is not going to be run on the same z/OS image as the SOAP client. The API that CA JARS will use to send messages to the SOAP client can find the SOAP client via CCI as long as it is set up. There is no setup needed in CA JARS to define a CCI pathway to the SOAP client.

CA JARS will open a request for all abend conditions encountered that appear to be a product failure. Conditions such as invalid input data or resource shortages will not cause CA Service Desk requests to be opened. The following z/OS abnormal completion codes will not generate a CA Service Desk request:

- 0C7
- 0C9
- *22
- *37

Note that CA JARS does not terminate if an 0C7 or 0C9 abend occurs. These are almost always caused by data errors in the input SMF data, and CA JARS recovers from the condition and continues normal operation. If this error does occur, the record is snapped to print (provided the snap limit has not been reached).

If an asterisk is part of the completion code, CA JARS will not open a request regardless of the code.

Configuration

If you have only one SOAP client and only one instance of CA Service Desk, CA Service Desk information can be configured within the SOAP client. However, if you want to route CA JARS requests to a CA Service Desk other than the one indicated in the SOAP client, you must provide some definitions in a data set that is allocated to the CAISDPRM DDNAME.

The definitions are coded as keyword=value pairs. Keywords are placed in column 1 in the file and are followed by an equal sign (=) and value. Comments have an asterisk (*) in column 1. All characters after the first blank character on a line are also considered to be comments. All character fields are case-sensitive, including the keywords themselves. For example, the UID= keyword is valid, while a keyword of uid= is invalid and will be ignored.

The following keywords are supported:

PRI

This tells CA JARS what priority to assign to requests. Valid values range from 1 to 5. The default value is 3.

URL

This is the web address of the CA Service Desk web service for the instance of CA Service Desk where you want to route the request. The default is the value of SD_URL in the SOAP client settings.

UID

This is an alternate user ID (from the default in the SOAP client) that is used to access the CA Service Desk on the target machine. The default is the value of SD_UID in the SOAP client settings.

Note: This will not be the value of the user ID in the request that is opened.

PSW

This is the password for the alternate user ID above. It is required if an alternate user ID is used. There is no default value. If SD_UID is being used as the user ID for the CA Service Desk, this parameter should be omitted; otherwise unpredictable results can occur.

If an invalid keyword is placed in the data set, it is ignored. There will be no error messages indicating an invalid keyword. Since one of the keywords might be a password, the parameters are not printed when they are processed.

A sample CAIDSPRM data set looks like this:

```
//CAISDPRM DD *
*
* CA Service Desk Optional Parameters
*
URL=servicedesk01.yourcompany.com/usd_ws/usd_ws.asmx
UID=servicedesk01
PSW=psw12345
PRI=2
```

Reporting

If an abend occurs in CA JARS that is not handled by CA JARS and if CA JARS is set up to use CA Service Desk, a request will be opened in CA Service Desk. The description is the primary source of information about the abend and looks like this:

```
ABEND0C1 IN CA JARS rnn JSIRPG20 00007FB2. yyyy0614 140210
PSW: 078D000000067A62 0001 0002 JSIRPG20 00007FB2
GENERAL REGISTERS (0-15):
 0000000000000000 00000000000BF3C6
00000000A00640FE 0000000000058B42
0000000000000005 0000000000000000
0000000000009C18 0000000000000024
0000000000067340 00000000000BF3C6
0000000000069340 0000000000071B58
0000000000068340 0000000000060484
00000000A00673AA 0000000000000000
INSTRUCTIONS NEAR ERROR (00067A5C): A4013000 0060A380 95E86768
Z/OS SYSTEM INFORMATION: XE44 XE44 SP7.0.4
JOB INFORMATION: JOB02J JOB05803 0046 yyyy0614 140207 007D1588
PRODUCT INFO: CA JARS rnn SPn
```

Each line displays information about the error and is useful in diagnosis, either by your site or by CA Technical Support. The lines have the following meaning:

Line Numbers	Meaning
1	This line indicates the error that was encountered. The abend code (ABENDxxx for system abends in hex, USERnnnn for user abends in decimal) and load module name and offset are included, as well as the date (YYYYMMDD) and time (HHMMSS - 24 hour clock).

Line Numbers	Meaning
2	This is the PSW at the time of the error, followed by the interrupt code, the instruction length code, and a repeat of the load module name and offset.
3	This line indicates the next 8 lines are the registers, displayed in 64-bit format.
4 to 11	These lines are the 64-bit general registers at the time of the abend. The first line has registers 0 and 1, the second has registers 2 and 3 and so on, until line 11, which has registers 14 and 15.
12	This line will show instructions near the address in the PSW if they are available in memory. The address in parentheses in front of the instructions is the address of the first byte of the printed instructions. Typically, this is 6 bytes prior to the address in the PSW.
13	This line displays the system name where the error occurred, as well as the SMF ID and the version of the operating system from the CVT prefix.
14	This line indicates the job name, the job number, the address space ID, the date and time the job started running, and the address of the TCB that abended.
15	The final line indicates the product name, version, and service pack level.

A summary line at the end is the same as the first line of the description. This line will also appear on the list of requests.

Chapter 5: User Exits

This chapter documents the detail linkage conventions for interfacing to five user exits. The general flow of CA JARS products and where these exits fit into that flow were explained in Chapter 1 of this guide.

To accommodate installation accounting and reporting requirements not supported by CA JARS standard features, provision is made for user exit routines to augment or modify the actions normally taken. These user exit routines should be written in Assembler Language. This chapter describes the parameters passed to exit routines at each of the following five phases in processing logic:

- Processing of SMF input files (input exit).
- Creation of the working account file (account exit).
- Processing of the working account file prior to the application of billing algorithms (exit-1 exit).
- Processing of the working account file after the application of billing algorithms (exit-2 exit).
- Processing of user-defined reports prior to formatting a report print line (exit-3 exit).

Exit routines must reside in a LNKLST library or in the library in which CA JARS resides, as indicated by the STEPLIB DD statement, or in a library concatenated to the STEPLIB DD statement.

It is not necessary to link edit an exit routine with any part of this product. The exit routines are compiled and linked independently. CA JARS uses a *late binding* technique of exit name specification. This is implemented by permitting exit name specification in the PARMS (input and account exits) and exits (exit-1, exit-2, and exit-3) statements at report program execution time. See the *User Guide* for the appropriate control statement syntax.

The user exits use standard IBM linkage conventions. Upon entry from CA JARS, the registers are as follows:

Register	Description
15	Address of exit routine's entry point
14	Return address
13	Address of register save area
1	Address of parameter list

- Each call to any exit point is identical, except the case of EOF on input. That is, there is no *first time* flag or signal to the exit.
- Note that these exits do not have to be reentrant (RENT) or reusable (REUS).
- User exits should be link edited to a user load library to ensure persistence when future installation or deployment is performed.

The user exit must be coded and assembled using standard assembly language procedures. The following is for illustration purposes only and should be modified according to your installation requirements.

```
//JOBNAME JOB ,??????,CLASS=?,MSGCLASS=?
//****************************************************************
//* DOC:  This job assembles and links an exit into your loadlib *
//****************************************************************
//STEP1  EXEC PGM=ASMBLR,PARM='DECK,NOOBJECT'
//SYSPRINT DD  SYSOUT=*
//SYSUT1   DD  DSN=&.&SYSUT1.,UNIT=SYSDA,SPACE=(1700,(600,100))
//SYSUT2   DD  DSN=&.&SYSUT2.,UNIT=SYSDA,SPACE=(1700,(300,50))
//SYSUT3   DD  DSN=&.&SYSUT3.,UNIT=SYSDA,SPACE=(1700,(300,50))
//SYSPUNCH DD DSN=&.&TEMP(????EXIT).,
//            DISP=(NEW,PASS),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//            UNIT=SYSDA,
//            SPACE=(TRK,(1,1,1))
//SYSLIB   DD DSN=SYS1.MACLIB,DISP=SHR
//         DD DSN=CAI.CAJRSAMP,DISP=SHR
//SYSIN    DD DSN=CAI.youruserlib(????EXIT),DISP=SHR
//*
//LINK1  EXEC PGM=IEWL,PARM='LET,LIST,NCAL,XREF,MAP'
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DSN=&.&SYSUT1.,UNIT=SYSDA,SPACE=(1024,(100,10))
//SYSLMOD  DD DSN=your.loadlib,DISP=SHR
//OBJECT   DD DSN=&.&TEMP.,DISP=(OLD,DELETE)
//SYSLIN   DD *
 INCLUDE   OBJECT(????EXIT)
 NAME      ????EXIT(R)
/*
```

This section contains the following topics:

Input Exit

The input exit, if activated, is to pass each SMF record as it is read from the CAIJSSMF file (accepts only SMF records). This exit may modify, delete, or insert records prior to the editing, validating, and formatting logic applied by CA JARS. This exit is invoked by placing the load module name in positions 43-50 of the CA JARS PARMS statement.

The parameter list passed to the input exit routine has the following format:

Word	Description
1	Address of the SMF record about to be processed by CA JARS (pointing to the 4-byte Record Descriptor Word (RDW))
2	Address of a 1-character disposition indicator
3	Address of a 1-character insertion indicator
4	Address of user data area

A value of EBCDIC blank (hex '40') in the disposition indicator on entry to the input exit indicates that an SMF record is available to be processed. If the exit needs to modify or extract information from the record, it can do so at this time.

If the modifications include a change to the record length, the exit must move the record to its own work area, make the required modifications to the record in the work area, then store the address of the modified record back into the first word of the parameter list before returning control to CA JARS.

For the exit to delete the record from further processing, it must set the disposition indicator to any nonblank value before returning control to CA JARS. In order to insert records into the input stream, the exit must set the insertion indicator to any nonblank value, and store the address of the record to be inserted in the first word of the parameter list before returning control to CA JARS.

When the user exit is next invoked, after the inserted record has been processed, it will again receive a pointer to the original SMF record, which it can now choose to process or delete, or it can continue inserting records of its own. (Inserting SMF records requires that the exit be written in Assembler.)

The user data area pointed to by the fourth parameter is a 26-byte area with the following format:

Position	Description
1-16	16-byte character field (initialized to spaces)
17-21	5-byte packed decimal integer field (initialized to zero)

Position	Description
22-26	5-byte packed decimal field interpreted as time in hours to five decimal places (initialized to zero)

While this parameter is always available to the exit routine, its contents will be of value only when one of the following SMF record types has been made available to the exit routine:

SMF Record Number	Description
30	(subtype 2) Interval
30	(subtype 3) Interval End
30	(subtype 4) Step/Session End
30	(subtype 5) Job End
48	RJE Session Logoff

This area is appended to the transmittal (work) records created from each of these SMF record types and will be placed in the corresponding fields of the job or step accounting records. Note that if job level user fields are not changed from their initial values, the numeric fields from the step level records will be accumulated to the job account record, which requires that the packed decimal fields contain valid data.

In the case of one-step jobs, the user fields from the step level will be used for the one-step job account record.

A value of high-values (hex 'FF') in the disposition indicator on entry to the input exit indicates that end-of-file has been reached on SMFINP. No record is available for processing at this point, nor can any additional records be inserted. The user should take this opportunity to have the input exit prepare totals, generate user-formatted reports, close files, and so forth.

An account code in a z/OS job statement, for example, (1234,ABCD,X,Y,Z), is stored in an SMF record as a series of pairs consisting of one byte binary length followed by the account code element. From this example the data would be stored as:

x'04'1234 x'04'ABCD x'01'X x'01'Y x'01'Z

The space between pairs has been added for readability. The notation x'nn' indicates a one byte binary field, in hexadecimal representation. When the SMF record is processed, the length fields are removed from the data. In this example the account codes would be stored in the JARS STEP, JOB, or EXTDATA record as 1234ABCDXYZ.

Sample input exit CAJRDEXT can be used to restore the comma delimiters to account codes in the SMF record before the record is processed.

After APPLY processing, macro CAJRDEXT will be in CAJRSAMP. CAJRDEXT is a sample input exit, and can be considered as an option for CA JARS. This input exit, if activated, processes each SMF record type 4, 5, 20, 26, and 30 from the CAIJSSMF file which accepts only SMF records. The sample exit is invoked by:

1. Assembling and link-editing CAJRSAMP member CAJRDEXT.
2. Adding to the STEPLIB concatenation list the load library containing the resultant load module.
3. Placing the load module name (CAJRDEXT) in positions 43-50 of the PARMS control statement.

Note: Input Exit CAJRDEXT is written as non-reentrant code, and must be link-edited with parms AC(0),NORENT,NOREUS.

Account Exit

The account exit, if activated, is passed to each account record just as it is about to be written to the account file. This exit may modify, delete, or insert records at this point. Note that extreme care must be exercised if records are being inserted since no further editing or validation operations will be performed before the record is written to the account file.

This exit name is coded in by positions 51-58 on the PARMS control statement.

The parameter list contains the following parameters:

Word	Description
1	Address of logical record currently being processed (following the RDW)
2	Address of a 1-character disposition indicator
3	Address of a 1-character insertion indicator

Each account record is made available to the user exit routine. The sequence in which the records are passed to the exit routine for each job is step records first, followed by the job record summarizing the accumulated step information. The disposition and insertion indicators work in the same manner as when employed by the input exit. This allows you to edit and select basic accounting records which will ultimately be used during the reporting phase.

Records are inserted in exactly the same fashion as described for an input exit with the following exception: the record to be inserted must be preceded by a valid RDW which contains the length of the record (including the four bytes for the RDW), but the address stored into the first word of the parameter list must point to the beginning of the record after the RDW.

The accumulation from the step records to the job record is accomplished prior to passing. Therefore, they must complete the mathematical roll up process from the step to the job records. Account exits are easier to code than input exits and are more powerful because:

1. The user is passed the formatted basic accounting record rather than all the raw records which make up a step accounting record.
2. The exit has access to CAIJSAIP and CAIJSHST.
3. Any changes made in this exit will affect the total run.

Typical uses of an account exit are to:

- Insert additional information into the user fields.
- Modify accounting information because of new standards set up at a client site.
- Create an account record to report on information CA JARS does not carry.

Note: CA JARS will not accept a summary file input with mixed record types. Job or Job/Step (level 6 or level 7) history records may not be processed simultaneously with summary records (level 1 to 5). Also, only level 6 or level 7 records will be used to prepare System Use Reports.

See the *User Guide* for the layout of the basic accounting record.

To make coding easier, a DSECT that maps this record is distributed with this product and resides in CAJRMAC as member **JSIACSUM.**

A sample expansion of JSIACSUM is provided on the following pages.

```
                                    449          JSIACSUM A,RDW
                                    450+***********************************************************************
                                    451+*                                                                     *
                                    452+*      A C C O U N T - S U M M A R Y   R E C O R D                    *
                                    453+*                                                                     *
                                    454+***********************************************************************
                                    455+*
000000                              456+JARDW     DS    0XL4           RECORD DESCRIPTOR WORD.                  01-00016
000000                              457+JALRECL   DS    HL2'0'         RECORD LENGTH.                           01-00017
_000002                              458+          DS    HL2'0'         ZEROS.                                   01-00018
000004                              460+JACPUID   DS    CL1' '    57   CPU ID.                                  01-00022
                                    461+*
000005                              462+JAFILEID  DS    CL1' '         FILE IDENTIFIER (A=ACCOUNT, 1-7=HISTORY) 01-00024
                            000C1   463+JAACCT    EQU   C'A'                                                    01-00025
                            000F1   464+JALVL1    EQU   C'1'                                                    01-00026
                            000F2   465+JALVL2    EQU   C'2'                                                    01-00027
                            000F3   466+JALVL3    EQU   C'3'                                                    01-00028
                            000F4   467+JALVL4    EQU   C'4'                                                    01-00034
                            000F5   468+JALVL5    EQU   C'5'                                                    01-00035
                            000F6   469+JAJOB     EQU   C'6'                                                    01-00036
                            000F7   470+JAJOBSTP  EQU   C'7'                                                    01-00037
                                    471+*
000006                              472+JAFILEVL  DS    XL1'4C'        FILE VERSION LEVEL.                      01-00040
                            0004C   473+JACURLVL  EQU   X'4C'                                                   01-00041
                            0004C   474+JALEVL40  EQU   X'4C'                                                   01-00042
                            0002C   475+JALEVL30  EQU   X'2C'                                                   01-00043
                            0001C   476+JALEVL27  EQU   X'1C'                                                   01-00044
                                    477+*
000007                              478+JATIEBRK  DS    0CL25          SORT TIE BREAKER                         01-00046
000007                              479+JAJOBLOG  DS    0CL20          JOB LOG                                  01-00047
000007                              480+JARDRSDT  DS    CL6' '    60   READER START DATE  YYMMDD                01-00048
00000D                              481+JARDRSTM  DS    CL6' '    61   READER START TIME  HHMMSS                01-00049
000013                              482+JAJOBNM   DS    CL8' '    04,  JOB NAME                                 01-00050
_                                    483+*                          G8   RJE LINE NAME
00001B                              485+JASJIND   DS    0CL2           STEP/JOB INDICATORS.                     01-00053
00001B                              486+JASTEPI   DS    CL1' '         STEP INDICATOR.                          01-00054
                            000F1   487+JAISSTEP  EQU   C'1'             IS A STEP RECORD                       01-00055
 00001C                              488+JAJOBI    DS    CL1' '         JOB INDICATOR.                           01-00056
```

```
_                                                                                                               PAGE   11
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                             ASM H V 02 11.49 02/28/98
                            000F1   489+JAISJOB   EQU   C'1'             IS A JOB RECORD                        01-00057
                                    490+*
00001D                              491+JASTEPNO  DS    XL2'0000' 56   NUMBER OF STEPS/STEP NUMBER.             01-00059
                                    492+*                         XH
00001F                              493+JAPROCID  DS    CL1' '    03   RECORD TYPE (I.E. PROCESSING ID).        01-00061
                            000C3   494+JATYPEC   EQU   C'C'             CICS                                   01-00062
                            000C4   495+JATYPED   EQU   C'D'             DOS                                    01-00063
                            000C8   496+JATYPEH   EQU   C'H'             HISTORY LEVEL SUMMARY FIELDS           01-00064
                            000D9   497+JATYPER   EQU   C'R'             RJE RECORD                             01-00065
                            000E2   498+JATYPES   EQU   C'S'             OS                                     01-00066
                            000E3   499+JATYPET   EQU   C'T'             TSO                                    01-00067
                            000E4   500+JATYPEU   EQU   C'U'             UNIT-RECORD DEVICE RECORD              01-00068
                            00060   501+JATYPECR  EQU   C'-'             CREDIT RECORD                          01-00069
                            0004E   502+JATYPEDB  EQU   C'+'             DEBIT RECORD                           01-00070
                            0005C   503+JATYPEBD  EQU   C'*'             BUDGET RECORD                          01-00071
                            0007C   504+JATYPEI   EQU   C'@'             IPL RECORD                             01-00072
                            0007B   505+JATYPEW   EQU   C'#'             OS WAIT TIME, MVS     WAIT TIME RECORDS    01-00073
                            0006F   506+JATYPEL   EQU   C'?'             SMF DATA LOST RECORD                   01-00074
                            0007A   507+JATYPEEX  EQU   C':'             EXTDATA RECORD                         01-00075
                                    508+*
                                    509+*                                                *SYSTEM RECORD ONLY
                                    510+*
000020                              511+JAETIME   DS    PL6'0'         ELAPSED TIME                             01-00079
000026                              512+JADVAL    DS    PL6'0'         DATA VALUE                               01-00080
00002C                              513+JALEFLG   DS    C' '           LEGITIMATE ACCOUNT RECORD FLAG           01-00081
00002D                              514+JAWAIT    DS    16PL6'0'       INDIVIDUAL ENGINE WAIT TIMES             01-00082
00008D                              515+JAUPDN    DS    16X'00'        ENGINE UP/DOWN: BIT 7 ONLINE, AND OK     01-00083
00009D                      00020   516+          ORG   JAETIME                                                 01-00084
                                    517+*
                                    518+*                                               *ACCOUNT RECORD ONLY
                                    519+*
000020                              520+JAUSERID  DS    CL8' '    10,  USER IDENTIFICATION                      01-00088
                                    521+*                         G6   RJE PASSWORD
000028                              522+JASTARTT  DS    CL6' '    12   START TIME    HHMMSS                     01-00090
00002E                              523+JASTARTD  DS    CL6' '    06   START DATE    YYMMDD                     01-00091
000034                              524+JAJOBCLS  DS    CL2' '    07   JOB CLASS                                01-00092
000036                              525+JASTOPTM  DS    CL6' '    13   STOP TIME     HHMMSS                     01-00093
00003C                              526+JAPGMRNM  DS    0CL20     58   PROGRAMMER'S NAME                        01-00094
00003C                              527+JAPGMRN1  DS    CL16' '                                                 01-00095
00004C                              528+JAPGMRN2  DS    CL4' '                                                  01-00096
                                    529+*
000050                              530+JA$ACC$   DS    CL16' '   08   ACCOUNTING FIELDS                        01-00098
                                    531+*
000060                              532+JACOMPCD  DS    CL4' '    09   COMPLETION CODE                          01-00100
000064                              533+JATERMID  DS    CL2' '    59   TERMINATION INDICATOR                    01-00101
                                    534+*
000066                              535+JASTEPNM  DS    CL8' '    11,  STEP NAME                                01-00103
                                    536+*                         G7   RJE REMOTE NAME
00006E                              537+JAPGMNM   DS    CL8' '    05   PROGRAM NAME/MODULE NAME                 01-00105
                                    538+*
000076                              539+JACOREAL  DS    PL4'0'    22   CORE ALLOCATED                           01-00107
00007A                              540+JACOREUS  DS    PL4'0'    21   CORE USED                                01-00108
00007E                              541+JACOREPU  DS    PL3'0'    23   PERCENT OF UNUSED CORE        (999V99)   01-00109
                                    542+*
 000081                              543+JAJOBNO   DS    CL5' '    14,  JOB NUMBER   (EXCEPT FOR TYPE H RECORDS) 01-00111
```

```
                                                              PAGE   12
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                    ASM H V 02 11.49 02/28/98
                                    544+*                         B4   TSU NUMBER
000086                      00081   545+          ORG   JAJOBNO                                              01-00113
000081                              546+          DS    CL1            RESERVED             ****************   01-00114
000082                              547+JATSOSES  DS    XL2       XG   # OF TSO SESSIONS     * TYPE H RECORDS   01-00115
000084                              548+JABATCH   DS    XL2       XF   # OF BATCH JOBS       ****************   01-00116
                                    549+*
000086                              550+JAPRTY    DS    CL2' '    15   PRIORITY                                  01-00118
                                    551+*
000088                              552+JACARDRD  DS    PL6'0'    18   CARDS READ                                01-00120
00008E                              553+JALINEPN  DS    PL6'0'    E5   STANDARD-LINES-PRINTED.                   01-00121
000094                              554+JALINEPS  DS    PL6'0'         LINES PRINTED - SPECIAL                   01-00122
00009A                              555+JACARDPU  DS    PL6'0'    20   CARDS PUNCHED                             01-00123
0000A0                              556+JAFORMPT  DS    CL4' '    16   PRINT FORMS ID                            01-00124
0000A4                              557+JAFORMPU  DS    CL4' '    17   PUNCH FORMS ID                            01-00125
                                    558+*
0000A8                              559+JAIDLETM  DS    PL6'0'    24   SETUP/IDLE TIME  HOURS   (999999V99999)  01-00127
0000AE                              560+JAELAPTM  DS    PL6'0'    25   ELAPSED TIME     HOURS   (999999V99999)  01-00128
                                    561+*                         65
                                    562+*                         90
0000B4                              563+JACPUTM   DS    PL6'0'    26   CPU TIME         HOURS   (999999V99999)  01-00131
                                    564+*                         66
                                    565+*                         91
0000BA                              566+JAOVHDTM  DS    PL6'0'    27   OVERHEAD TIME    HOURS   (999999V99999)  01-00134
                                    567+*                         67
0000C0                              568+JAWAITTM  DS    PL6'0'    28   WAIT TIME        HOURS   (999999V99999)  01-00136
                                    569+*                         68
0000C6                              570+JARDRQTM  DS    CL6' '    62   READER QUEUE TIME    HHMMSS              01-00138
0000CC                              571+JAWTRQTM  DS    CL6' '    63   WRITER QUEUE TIME    HHMMSS              01-00139
0000D2                              572+JATURNTM  DS    CL6' '    64   TURNAROUND TIME      HHMMSS              01-00140
                                    573+*
                                    574+* DEVICE EXCP COUNTS BY DEVICE TYPE:
                                    575+*
0000D8                              576+JARDRIOC  DS    PL7'0'    33,  READER I/O COUNT                          01-00144
                                    577+*                         F8   RJE EXCP-COUNT
0000DF                              578+JAPRTIOC  DS    PL7'0'    34,  PRINTER I/O COUNT                         01-00146
                                    579+*                         G1   RJE NAKS
0000E6                              580+JAPUNIOC  DS    PL7'0'    35,  PUNCH I/O COUNT                           01-00148
                                    581+*                         G3   RJE DATA-CHECKS
0000ED                              582+JATAPIOC  DS    PL7'0'    37,  TAPE I/O COUNT                            01-00150
                                    583+*                         G5   RJE # INV-PASSWORDS
0000F4                              584+JADSKIOC  DS    PL7'0'    38   DISK I/O COUNT                            01-00152
0000FB                              585+JAOTHIOC  DS    PL7'0'    39,  OTHER I/O COUNT                           01-00153
                                    586+*                         F9   RJE OTHER-ERRORS
000102                              587+JATOTIOC  DS    PL7'0'    40   TOTAL COUNT OF ALL I/O                    01-00155
000109                              588+JAINDXIO  DS    PL4'0'    78   I/O INDEX                                 01-00156
                                    589+*
00010D                              590+JAAPPCTY  DS    0CL1' '   72   APPC TYPE                                 01-00158
                            000C9   591+JAINBND   EQU   C'I'           INBOUND                                   01-00159
                            000D6   592+JAOUTBND  EQU   C'O'           OUTBOUND                                  01-00160
                            000E3   593+JAAPPX    EQU   C'T'           TRANSACTION                               01-00161
                                    594+*
00010D                              595+JAMEMORY  DS    CL1' '    72   STORAGE INDICATOR                         01-00163
                            000D9   596+JAREAL    EQU   C'R'           REAL "R"     STORAGE                      01-00164
                            000E5   597+JAVIRT    EQU   C'V'           VIRTUAL "V"  STORAGE                      01-00165
                                    598+*
```

```
_                                                                                                               PAGE   13
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                     ASM H V 02 11.49 02/28/98
00010E                              599+JAPAGEIN  DS    PL7'0'    73   PAGE-IN COUNT                            01-00167
000115                              600+JAPAGEOT  DS    PL7'0'    75   PAGE-OUT COUNT                           01-00168
00011C                              601+JAPAGETL  DS    PL7'0'    76   TOTAL PAGING COUNT                       01-00169
000123                              602+JAPAGERT  DS    PL4'0'    77   CPU PAGING RATE=                         01-00170
                                    603+*                    (PAGE-TOTAL/CPU-TIME) * 3600 = (PAGES/SECOND)
000127                              604+JAPAGEEL  DS    PL4'0'    76   ELAPSED PAGING RATE=                     01-00172
                                    605+*                    (PAGE-TOTAL/ELAPSED-TIME) * 60 = (PAGES/MINUTE)
                                    606+*
00012B                              607+JAGROUP1  DS    CL1' '    84  GROUP CODE #1                             01-00175
00012C                              608+JAGROUP2  DS    CL1' '    85  GROUP CODE #2                             01-00176
00012D                              609+JAGROUP3  DS    CL1' '    86  GROUP CODE #3                             01-00177
                                    610+*
00012E                              611+JAJOBTYP  DS    0CL1'  '       JOB TYPE                                 01-00179
                            000C1   612+JAJAPPC   EQU   C'A'                APPC                                01-00180
                            000E2   613+JAJSNA    EQU   C'S'                SNA RJE                             01-00181
                            000C2   614+JAJBSC    EQU   C'B'                BSC RJE                             01-00182
                            000E2   615+JAJSTC    EQU   C'S'                STC                                 01-00183
                            000D1   616+JAJJOB    EQU   C'J'                JOB                                 01-00184
                            000E3   617+JAJTSU    EQU   C'T'                TSU                                 01-00185
                            0006F   618+JAJOTH    EQU   C'?'                OTHER                               01-00186
                                    619+*
00012E                              620+JAPARTID  DS    CL2' '    87   PARTITION ID                             01-00188
000130                              621+JAIDEVNM  DS    CL8' '         INPUT DEVICE NAME                        01-00189
000138                              622+JARDRDUR  DS    PL6'0'    88  READER DURATION    HOURS   (999999V99999) 01-00190
00013E                              623+JAWTRDUR  DS    PL6'0'    89  WRITER DURATION    HOURS   (999999V99999) 01-00191
                                    624+*
000144                              625+JANSWAPS  DS    PL7'0'    A1   NUMBER OF SWAPS                          01-00193
00014B                              626+JASPAGEI  DS    PL7'0'    A2   SWAP PAGES-IN                            01-00194
000152                              627+JASPAGEO  DS    PL7'0'    A3   SWAP PAGES-OUT                           01-00195
000159                              628+JASPAGET  DS    PL7'0'    A4   TOTAL SWAP PAGES                         01-00196
                                    629+*
                                    630+* DEVICE COUNTS BY DEVICE TYPE:
                                    631+*
000160                              632+JARDRUSE  DS    PL2'0'    92   NUMBER OF READERS USED                   01-00200
000162                              633+JAPRTUSE  DS    PL2'0'    93   NUMBER OF PRINTERS USED                  01-00201
000164                              634+JAPUNUSE  DS    PL2'0'    94   NUMBER OF PUNCHES USED                   01-00202
000166                              635+JATAPUSE  DS    PL2'0'    95   NUMBER OF TAPES USED                     01-00203
000168                              636+JADSKUSE  DS    PL2'0'    96   NUMBER OF DISKS USED                     01-00204
00016A                              637+JAOTHUSE  DS    PL2'0'    98   NUMBER OF OTHER DEVICES USED             01-00205
00016C                              638+JAPVTUSE  DS    PL2'0'    97   NUMBER OF PRIVATE DISKS USED             01-00206
00016E                              639+JATOTUSE  DS    PL2'0'    32   TOTAL  OF ALL DEVICES USED               01-00207
                                    640+*
000170                              641+JATPUTS   DS    PL4'0'    A5   TPUTS                                    01-00209
000174                              642+JATGETS   DS    PL4'0'    A6   TGETS                                    01-00210
                                    643+*
000178                              644+JAACTVTM  DS    PL6'0'    A7,  ACTIVE TIME      HOURS   (999999V99999)  01-00212
                                    645+*                         A8,
                                    646+*                         A9
00017E                              647+JACONNTM  DS    PL6'0'    B0,  CONNECT TIME     HOURS   (999999V99999)  01-00215
                                    648+*                         B1,
                                    649+*                         B2
000184                              650+JAALDLTM  DS    PL6'0'    B8,  ALLOC. DEL. TIME HOURS   (999999V99999)  01-00218
                                    651+*                         B9,
                                    652+*                         C0
 00018A                              653+JARESDTM  DS    PL6'0'    C1,  RESIDENT TIME    HOURS   (999999V99999)  01-00222
```

```
_                                                                                                                PAGE   14
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                         ASM H V 02 11.49 02/28/98
                                     654+*                         C2,
                                     655+*                         C3
 000190                              656+JACPUSRB  DS    PL6'0'    C4,  CPU TIME (SRB)   HOURS   (999999V99999)  01-00225
                                     657+*                         C5,
                                     658+*                         C6
 000196                              659+JACPUTCB  DS    PL6'0'    C7,  CPU TIME (TCB)   HOURS   (999999V99999)  01-00228
                                     660+*                         C8,
                                     661+*                         C9
 00019C                              662+JASVCUNT  DS    PL6'0'    D1   SERVICE UNITS                            01-00231
 0001A2                              663+JASVCRT   DS    PL3'0'    D2   SERVICE RATE=              (999V99)      01-00232
                                     664+*             (SERVICE-UNITS/ACTUAL-TIME) * 3600 = SERVICE-UNITS/SECOND
 0001A5                              665+JACPUIDX  DS    PL2'0'    D0   CPU INDEX =                  (9V99)      01-00234
                                     666+*                         (CPUSRB/CPU-TIME) * 100 = SRB % OF TOTAL
 0001A7                              667+JAPAGESS  DS    PL6'0'    D3   PAGE SECONDS                             01-00236
 0001AD                              668+JAPERGRP  DS    CL3'0'    D4   PERFORMANCE GROUP                        01-00237
 0001B0                              669+JAIROUTE  DS    CL3' '    B6   INPUT ROUTE CODE               ZZ9       01-00238
 0001B3                              670+JAPROUTE  DS    CL3' '    B7   PRINT ROUTE CODE               ZZ9       01-00239
                                     671+*
                                     672+* JOB RESTART FLAGS:
                                     673+*
 0001B6                              674+JAJOBRQI  DS    0CL2      B5   JOB REQUEUED INDICATOR                   01-00243
 0001B6                              675+JAWRSTRT  DS    CL1' '         '1'=JOB WAS RESTARTED.                   01-00244
 0001B7                              676+JAIRSTRT  DS    CL1' '         '1'=JOB IS A RESTART.                    01-00245
 0001B8                              677+JADTIND   DS    CL1'0'    D5,  DAY-OF-WEEK-IND.                         01-00247
                                     678+*                         D6
 0001B9                              679+JAGROUP4  DS    CL1' '    D7   GROUP-CODE-4.                            01-00249
 0001BA                              680+JAGROUP5  DS    CL1' '    D8   GROUP-CODE-5.                            01-00250
 0001BB                              681+JAGROUP6  DS    CL1' '    D9   GROUP-CODE-6.                            01-00251
 0001BC                              682+JAFORMCT  DS    PL2'0'    E0   #FORMS ENTRIES.                          01-00252
 0001BE                              683+JASOUTGP  DS    0CL22                                                   01-00253
 0001BE                              684+JAPRTCLS  DS    CL1' '    E1   PRINT-CLASS.                             01-00254
 0001BF                              685+JASYSOTP  DS    CL1'1'    E2   SYSOUT-TYPE (P=PRT,C=PUN)                01-00255
 0001C0                              686+JAOPRICD  DS    CL1' '    E3   OPR-INTV-CD.                             01-00256
 0001C1                              687+JAOUTDNM  DS    CL8' '    E4   OUTPUT-DEVICE-NAME.                      01-00257
 0001C9                              688+JALINEPT  DS    PL6'0'    19   LINES PRINTED                            01-00258
 0001CF                              689+JAPAGEPN  DS    PL5'0'    E6   STANDARD-PAGES-PRINTED.                  01-00259
 0001D4                              690+JAPAGEPS  DS    PL5'0'    E7   SPECIAL-PAGES-PRINTED.                   01-00260
 0001D9                              691+JAPAGEPT  DS    PL5'0'    E8   TOTAL-PAGES-PRINTED.                     01-00261
 0001DE                              692+JATAPMTS  DS    PL4'0'    F0   #TAPES-MOUNTED.                          01-00262
 0001E2                              693+JADSKMTS  DS    PL4'0'    F1   #DISKS-MOUNTED.                          01-00263
 0001E6                              694+JAABSORP  DS    PL4'0'    E9   ABSORPTION-RATE.                         01-00264
 0001EA                              695+JAACCT2   DS    CL16' '   F2   2ND 16 BYTE ACCT-FIELD.                  01-00265
 0001FA                              696+JAACCT3   DS    CL16' '   F3   3RD 16 BYTE ACCT-FIELD.                  01-00266
 00020A                              697+JAUSRCHR  DS    CL16' '   F4   USER 16 BYTE CHAR-FLD.                   01-00267
 00021A                              698+JAUSRCT   DS    PL5'0'    F5   USER-INTEGER-CNT-FLD.                    01-00268
 00021F                              699+JAUSRTM   DS    PL5'0'    F6   USER TIME-FLD. S9999V99999 HOURS.        01-00269
 000224                              700+JARSRVD1  DS    CL6' '         6 BYTE-RESERVED AREA.                    01-00270
                                     701+*
                             0022A   702+JAENDCOM  EQU   *              END OF COMMON AREA                       01-00276
                             00226   703+JACOMLEN  EQU   *-JACPUID      COMMON AREA LENGTH                       01-00277
                                     704+*
                                     705+*   S U M M A R Y    RECORD ONLY FIELDS
                                     706+*
 00022A                              707+JAPROCTM  DS    PL6'0'    29,  PROC. TIME     HOURS    (999999V99999)   01-00281
                                     708+*                         69
```

```
_                                                                                                               PAGE   15
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                     ASM H V 02 11.49 02/28/98
□000230                              709+JAPROCHG  DS    PL6'0'    44   PROC. CHARGE   DOLLARS  (999999999V99)   01-00283
□000236                              710+JAIOTIME  DS    PL6'0'    30,  I/O TIME       HOURS    (999999V99999)   01-00284
□                                    711+*                         70
□00023C                              712+JAIOCHG   DS    PL6'0'    45   I/O CHARGE     DOLLARS  (999999999V99)   01-00286
□000242                              713+JAURCHG   DS    PL5'0'    54   U/R CHARGE     DOLLARS    (9999999V99)   01-00287
□000247                              714+JASETCHG  DS    PL5'0'    55   SET-UP CHARGE  DOLLARS    (9999999V99)   01-00288
□00024C                              715+JATOTCHG  DS    PL6'0'    46   TOTAL CHARGE   DOLLARS  (999999999V99)   01-00289
□000252                              716+JACHGSUF  DS    CL1' '         CHARGE SUFFIX                            01-00290
□000253                              717+JAADJTRT  DS    PL6'0'         RESERVED (ADJ. RATE)    (999999999V99)   01-00291
□000259                              718+JACONCHG  DS    PL5'0'    B3   CONNECT CHARGE DOLLARS    (9999999V99)   01-00293
□00025E                              719+JARJECHG  DS    PL5'0'    G9   RJE CHARGE     DOLLARS    (9999999V99)   01-00295
□000263                              720+JACLASCD  DS    PL1'0'         CLASS CODE                               01-00297
□                            00264   721+JAENDSUM  EQU   *                   END OF SUMMARY RECORDS ONLY FIELDS  01-00298
□                            0003A   722+JASUMLEN  EQU   *-JAPROCTM          LENGTH OF SUMMARY ONLY PORTION      01-00299
□                                    723+*
□                                    724+* SYSOUT TABLE (0-16 50-BYTE ENTRIES)           ### SUMMARY RECORD ###
□                                    725+* BASED ON 'J.FORMCT'
□                                    726+*
□000264                              727+JASSYSTB  DS    0CL50          50-BYTE-SYSOUT-TBL.                      01-00305
□000264                              728+JASSYSDT  DS    CL6' '         SYSOUT-START-DT.     YYMMDD              01-00306
□00026A                              729+JASSYSIT  DS    PL3'0'         SYSOUT-START-TIME.   SECONDS             01-00307
□00026D                              730+JASSYSET  DS    PL3'0'         SYSOUT-STOP-TIME.    SECONDS             01-00308
□000270                              731+JASSYSDR  DS    PL4'0'         SYSOUT-DURATION.     HOURS   99V99999    01-00309
□000274                              732+JASSYSGP  DS    0CL22          SYSOUT-GROUP-FLD.                        01-00310
□000274                              733+JASSYSCL  DS    CL1' '         SYSOUT-CLASS.                            01-00311
□000275                              734+JASSYSTP  DS    CL1' '         SYSOUT-TYPE. 1=STD PRT,2=SPCL PRT,       01-00312
□                                    735+*                                           3=STD PUN,4=SPCL PUN
□000276                              736+JASSYSIC  DS    CL1' '         SYSOUT-OP-INT-CD.                        01-00314
□000277                              737+JASSYSDN  DS    CL8' '         SYSOUT-OUTPUT-DEVICE-NAME.               01-00315
□00027F                              738+JASSYSLN  DS    PL6'0'         SYSOUT-LINES-PRINTED.                    01-00316
□000285                              739+JASSYSPG  DS    PL5'0'         SYSOUT-PAGES-PRINTED.                    01-00317
□00028A                              740+JASSYSFI  DS    CL4' '         SYSOUT-FORMS-ID.                         01-00318
□00028E                              741+JASSYSRC  DS    CL3' '         SYSOUT-ROUTE-CD.                         01-00319
□000291                              742+JASSYSCH  DS    PL5'0'         SYSOUT-CHARGE  DOLLARS    (9999999V99)   01-00320
□000296                              743+JASSYNX   DS    0CL50          NEXT SYSOUT-TABLE-ENTRY                  01-00321
□000296                      00264   744+          ORG   JAENDSUM                                                01-00322
□                                    745+*
□                                    746+* IO DEVICE TABLE (0-256 8-BYTE ENTRIES)        ### SUMMARY RECORD ###
□                                    747+* BASED ON THE TOTAL OF ALL DEVICES 'J.TOTUSE'
□                                    748+*
□000264                              749+JASDEVTB  DS    0CL8           IO DEVICE TABLE.                         01-00328
□                                    750+*
□000264                              751+JASDEVTY  DS    PL1'0'         DEVICE TYPE (1-7).                       01-00330
□                            0001C   752+JARDRDEV  EQU   X'1C'       P'1'    READER                              01-00331
□                            0002C   753+JAPRTDEV  EQU   X'2C'       P'2'    PRINTER                             01-00332
□                            0003C   754+JAPUNDEV  EQU   X'3C'       P'3'    PUNCH                               01-00333
□                            0004C   755+JATAPDEV  EQU   X'4C'       P'4'    TAPE                                01-00334
□                            0005C   756+JADSKDEV  EQU   X'5C'       P'5'    DISK                                01-00335
□                            0006C   757+JAOTHDEV  EQU   X'6C'       P'6'    OTHER                               01-00336
□                            0007C   758+JAPVTDEV  EQU   X'7C'       P'7'    PRIVATE DISK                        01-00337
□000265                              759+JASDEVAD  DS    CL3' '         CHANNEL/UNIT ADDRESS (EBCDIC).           01-00338
□000268                              760+JASDEVIO  DS    PL4'0'         EXCP COUNT FOR DEVICE.                   01-00339
□                                    761+*
□                                    762+*
                                     763+*   A C C O U N T   RECORD ONLY FIELDS
```

```
_                                                                                                               PAGE   16
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                     ASM H V 02 11.49 02/28/98
□                                    764+*
□00026C                      0022A   765+          ORG   JAENDCOM                                                01-00344
□                                    766+*
□                                    767+* SYSOUT TABLE (0-16 50-BYTE ENTRIES)           ### SUMMARY RECORD ###
□                                    768+* BASED ON 'J.FORMCT'
□                                    769+*
□                                    770+*
□00022A                              771+JAASYSTB  DS    0CL50          50-BYTE-SYSOUT-TBL.                      01-00351
□00022A                              772+JAASYSDT  DS    CL6' '         SYSOUT-START-DT.     YYMMDD              01-00352
□000230                              773+JAASYSIT  DS    PL3'0'         SYSOUT-START-TIME.   SECONDS             01-00353
□000233                              774+JAASYSET  DS    PL3'0'         SYSOUT-STOP-TIME.    SECONDS             01-00354
□000236                              775+JAASYSDR  DS    PL4'0'         SYSOUT-DURATION.     HOURS   99V99999    01-00355
□00023A                              776+JAASYSGP  DS    0CL22          SYSOUT-GROUP-FLD.                        01-00356
□00023A                              777+JAASYSCL  DS    CL1' '         SYSOUT-CLASS.                            01-00357
□00023B                              778+JAASYSTP  DS    CL1' '         SYSOUT-TYPE. 1=STD PRT,2=SPCL PRT,      X01-00358
□                                       +                                            3=STD PUN,4=SPCL PUN
□00023C                              779+JAASYSIC  DS    CL1' '         SYSOUT-OP-INT-CD.                        01-00359
□00023D                              780+JAASYSDN  DS    CL8' '         SYSOUT-OUTPUT-DEVICE-NAME.               01-00360
□000245                              781+JAASYSLN  DS    PL6'0'         SYSOUT-LINES-PRINTED.                    01-00361
□00024B                              782+JAASYSPG  DS    PL5'0'         SYSOUT-PAGES-PRINTED.                    01-00362
□000250                              783+JAASYSFI  DS    CL4' '         SYSOUT-FORMS-ID.                         01-00363
□000254                              784+JAASYSRC  DS    CL3' '         SYSOUT-ROUTE-CD.                         01-00364
□000257                              785+JAASYCH   DS    PL5'0'         SYSOUT-CHARGE  DOLLARS    (9999999V99)   01-00365
□00025C                              786+JAASYNX   DS    0CL50          NEXT SYSOUT-TABLE-ENTRY                  01-00366
□00025C                      0022A   787+          ORG   JAENDCOM                                                01-00367
□                                    788+*
□                                    789+* IO DEVICE TABLE (0-256 8-BYTE ENTRIES)        ### ACCOUNT RECORD ###
□                                    790+* BASED ON THE TOTAL OF ALL DEVICES 'J.TOTUSE'
□                                    791+*
□00022A                              792+JAADEVTB  DS    0CL8           IO DEVICE TABLE.                         01-00373
□                                    793+*
□00022A                              794+JAADEVTY  DS    PL1'0'         DEVICE TYPE (1-7).                       01-00375
□00022B                              795+JAADEVAD  DS    CL3' '         CHANNEL/UNIT ADDRESS (EBCDIC).           01-00376
□00022E                              796+JAADEVIO  DS    PL4'0'         EXCP COUNT FOR DEVICE.                   01-00377
□000232                      00296   797+          ORG                                                           01-00378
□                                    798+*
□                                    799+*   S P E C I A L    RECORD ONLY FIELDS     (POST SORT RECORD)
□                                    800+*
□000296                      00264   801+          ORG   JAENDSUM                                                01-00384
□                                    802+*
□000264                              803+JACNTL01  DS    0XL32      XC  WORK AREA FIELDS FOR "JSIRPG6"           01-00386
□000264                              804+JASUMJOB  DS    PL5'0'     14  TOTAL JOBS       ** SUMMARY **           01-00387
□000269                              805+JASUMSTP  DS    PL4'0'     56  TOTAL STEPS      ** SUMMARY **           01-00388
□00026D                              806+JASUMSES  DS    PL4'0'     H3  TOTAL SESSIONS   ** SUMMARY **           01-00389
□000271                              807+JASUMTSO  DS    PL5'0'     B4  TSO SESSIONS     ** SUMMARY **           01-00390
□000276                              808+JAURIOC   DS    PL5'0'     36  TOTAL UNIT RECORD I/O COUNT  (999999999) 01-00391
□00027B                              809+JATOTIME  DS    PL6'0'     66  TOTAL TIME       HOURS    (999999V99999) 01-00392
□000281                              810+JASELFLG  DS    XL1'00'    XR  SORT SELECTION REASON FLAG (JSIMOVEA)    01-00393
□                            00001   811+JASELSUM  EQU   X'01'            SELECTED FOR SUMMARY FILE CREATION     01-00394
□                            00010   812+JASELACM  EQU   X'10'            SELECTED FOR REPORT ACCUMULATION       01-00395
□                            00020   813+JASELPRT  EQU   X'20'            SELECTED FOR REPORT DETAIL PRINTING    01-00396
□000282                              814+JAPROCIB  DS    XL2'0000'  XP  PROCESS ID - CONVERTED FOR BIT TESTING   01-00397
□                                    815+*
□000284                              816+JADEV001  DS    PL4'0'     G0                                           01-00399
 000288                              817+JADEV002  DS    PL4'0'     G0                                           01-00400
```

```
_                                                                                                              PAGE   17
_  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                                    ASM H V 02 11.49 02/28/98
□00028C                              818+JADEV003  DS    PL4'0'     G0                                          01-00401
□000290                              819+JADEV004  DS    PL4'0'     G0                                          01-00402
□000294                              820+JADEV005  DS    PL4'0'     G0       EXPANDED  DEVADDR                  01-00403
□000298                              821+JADEV006  DS    PL4'0'     G0                                          01-00404
□00029C                              822+JADEV007  DS    PL4'0'     G0         ENTRIES   AREA                   01-00405
□0002A0                              823+JADEV008  DS    PL4'0'     G0                                          01-00406
□0002A4                              824+JADEV009  DS    PL4'0'     G0                                          01-00407
□0002A8                              825+JADEV010  DS    PL4'0'     G0                                          01-00408
□0002AC                              826+JADEV011  DS    PL4'0'     G0                                          01-00409
□0002B0                              827+JADEV012  DS    PL4'0'     G0                                          01-00410
□0002B4                              828+JADEV013  DS    PL4'0'     G0                                          01-00411
□0002B8                              829+JADEV014  DS    PL4'0'     G0                                          01-00412
□0002BC                              830+JADEV015  DS    PL4'0'     G0                                          01-00413
□0002C0                              831+JADEV016  DS    PL4'0'     G0                                          01-00414
□0002C4                              832+JADEV017  DS    PL4'0'     G0                                          01-00415
□0002C8                              833+JADEV018  DS    PL4'0'     G0                                          01-00416
□0002CC                              834+JADEV019  DS    PL4'0'     G0                                          01-00417
□0002D0                              835+JADEV020  DS    PL4'0'     G0                                          01-00418
□                                    836+*
□                            00014   837+JADEVCNT  EQU   20                 TOTAL SPECIAL DEVICE ENTRIES        01-00420
□                                    838+*
□0002D4                              839+JASORTKY  DS    0CL160     S0     SORT KEY EXPANSION AREAS             01-00422
□0002D4                              840+JASORTK1  DS    CL32' '    S1                                          01-00423
□0002F4                              841+JASORTK2  DS    CL32' '    S2                                          01-00424
□000314                              842+JASORTK3  DS    CL32' '    S3                                          01-00425
□000334                              843+JASORTK4  DS    CL32' '    S4                                          01-00426
□000354                              844+JASORTK5  DS    CL32' '    S5                                          01-00427
□                                    845+*
□000374                              846+JAJOB#    DS    CL5' '         NO. OF JOBS                             01-00429
□000379                              847+JATSU#    DS    CL5' '         NO. OF STEPS                            01-00430
□                                    848+*
□00037E                      0037E   849+          ORG                                                          01-00432
□                            0037E   850+JASPCLEN  EQU   *-JARDW        SPECIAL PORTION LENGTH                  01-00434
□                            0037E   851+JAMAXLEN  EQU   *-JARDW        MAXIMUM RECORD LENGTH                   01-00440
□00037E                      0037E   852+          ORG                                                          01-00445
```

Exits 1, 2, and 3

The records passed to exits 1 and 2 are created from the raw input data and step/job (level 6 and 7) history files. When passing account records created from CREDIT, DEBIT and BUDGET statements, these are in the format of one step jobs with the amount from the *statements* in the working storage area of *total charge.*

See the chapter "System Description" for a detailed explanation of where Exits 1, 2, and 3 fit into the processing flow of the CA JARS product.

Exit 1

This exit provides control to the user immediately following the grouping feature logic and prior to application of the accounting algorithm for each accounting record being processed. This allows the user to perform tests or apply unique algorithms to records reaching this phase of processing. Step records for a particular job will be passed to the user exit routine prior to passing the associated job record.

This exit name is coded in positions 10-17 on the EXITS control statement.

If CA JARS input includes historical data, this information will also be passed to the user exit routine following all newly created account records (from SMF or DOS accounting records) but prior to all old account records.

The parameter list for this exit is as follows:

Word	Description
1	Address of Set Code for report currently being generated
2	Address of accounting record currently being processed
3	Address of RATE statement information table
4	Address of PRIORITY statement information table
5	Address of working storage area
6	Address of disposition indicator to be set by user
7	Address of TSORATE statement information table
8	Address of FORMRATE statement information table
9	Address of RJERATE/SNARATE/APPCRATE statement information table

The RATE statement information table consists of a variable number of entries, with the last valid entry being followed by an entry containing low-values (x'00'). Each entry is basically in the same format as the RATE statement as follows:

Field Name	Field Length	Displacement
CPU Identification	1	0
Basic Processor Rate	4	1
Basic I/O Rate	3	5
Core Factor	4	8
Elapsed Time Factor	3	12
Total CPU Time Factor	3	15
SRB CPU Time Factor	3	18
TCB CPU Time Factor	3	21
Reader I/O Factor	3	24
Printer I/O Factor	3	27
Punch I/O Factor	3	30
Tape I/O Factor	3	33

	Field Name	Field Length	Displacement
	Disk I/O Factor	3	36
	Other I/O Factor	3	39
	Reader Rate	3	42
*	Printer Rate	3	45
*	Special Print Rate	3	48
*	Punch Rate	3	51
	Tape Allocation Charge	3	54
	Disk Allocation Charge	3	57
	Minimum Job Charge	4	60
	Maximum Step Rate	4	64
	Step Time Criteria	2	68
	Core Indicator	1	70

* Not used for z/OS jobs (FORMRATE used instead)

The TSORATE statement information table consists of a variable number of entries with the last entry being followed by an entry containing low-values (x'00'). Each entry is basically in the same format as the TSORATE statement as follows:

Field Name	Field Length	Displacement
CPU Identification	1	0
Basic Processor Rate	4	1
Basic I/O Rate	3	5
Core Factor	4	8
Connect Time Rate	4	12
Active Time Factor	3	16
Total CPU Time Factor	3	19
Reader I/O Factor	3	22
Printer I/O Factor	3	25
Punch I/O Factor	3	28
Tape I/O Factor	3	31
Disk I/O Factor	3	34

Field Name	Field Length	Displacement
Other I/O Factor	3	37
Filler	12	40
TPUT Rate	3	52
TGET Rate	3	55
Tape Allocation Charge	3	58
Disk Allocation Charge	3	61
Minimum Session Charge	4	64

The PRIORITY statement information table consists of a variable number of entries with the last entry being followed by an entry containing low-values (x'00'). Each entry is basically in the same format as the PRIORITY statement as follows:

Field Name	Field Length	Displacement
CPU Identification	1	0
Processor Charge Flag	1	1
I/O Charge Flag	1	2
U/R Charge Flag	1	3
Setup Charge Flag	1	4
Priority Factors	30	5
Class/Partition Factors	35	35

The FORMRATE statement information table consists of a variable number of entries with the last entry being followed by an entry containing low-values (x'00'). Each entry is in a format similar to the FORMRATE statement with numeric fields converted to packed decimal as follows:

Field Name	Field Length	Displacement	Format
CPU Identification	1	0	character
SYSOUT Class	1	1	character
Forms ID	4	2	character
Setup Charge	3	6	packed (999V99)
Line Rate	3	9	packed (999V99)

Field Name	Field Length	Displacement	Format
Page Rate	3	12	packed (999V99)
Charge Option	1	15	character

The RJERATE and SNARATE statement information table consists of a variable number of entries with the last entry being followed by an entry containing low-values (x'00'). Each entry is in a format similar to the RJERATE or SNARATE statement with numeric fields converted to packed decimal as follows:

Field Name	Field Length	Displacement	Format
CPU Identification	1	0	character
Line Name	8	1	character
Rate Type	1	9	character
Time Rate	4	10	packed (99999V99)
Record Rate	3	14	packed (999V99)
Connect Rate	4	17	packed (99999V99)
Transmission Rate	3	21	packed (999V99)
Minimum Session Charge	3	24	packed (999V99)
Job Name	8	27	character
User Information	16	35	character

Rate type is a 'B' for BSC rates and an 'S' for SNA rates.

When the record being processed is a summary record from the history file or account record, the working storage area will contain the time and charge values accumulated into that record when it was created. So, any alterations to data from a summary level history file must be made directly to the record and cannot reference the working storage area.

Field Name	Field Length	Displacement	Format
CPU Identification	1	0	character
TP Name	8	1	character
Rate Type	9	1	character

Field Name	Field Length	Displacement	Format
Batch Send Rate	3	10	packed (99V999)
Batch Receive Rate	3	13	packed (99V999)
Outbound Send Rate	3	16	packed (99V999)
Outbound Receive Rate	3	19	
Outbound Call Rate	3	22	
Inbound Send Rate	3	25	
Inbound Receive Rate	3	28	
Inbound Call Rate	3	31	
Transaction Send Rate	3	34	
Transaction Receive Rate	3	37	
Transaction Conversion Rate	3	40	
Transaction TCB Rate	3	43	(99999)
Transaction SRB Rate	3	46	(99999)
Transaction EXCP Rate	3	49	(99V999)
Transaction Device Connect	3	52	(99V999)

Rate type is an 'A' for APPC rates.

Only the RATE, TSORATE, FORMRATE, RJERATE, APPCRATE/ SNARATE, and PRIORITY statements required to generate a given report will be present in their respective tables during this execution phase. Table entries will be in the same order as the statements were read by CA JARS. The working storage area referenced in the parameter list contains the following fields:

Field Name	Length	Field Displacement	Mode	Notes
Processor Time	6	0	Packed	999999V99999C; hours
Processor Charge	6	6	"	999999999V99C; dollars
I/O Time	6	12	"	999999V99999C; hours
I/O Charge	6	18	"	999999999V99C; dollars
U/R Charge	5	24	"	9999999V99C; dollars
Setup Charge	5	29	"	9999999V99C; dollars
Total Charge	6	34	"	999999999V99C; dollars

Field Name	Length	Field Displacement	Mode	Notes
Charge Suffix	1	40	EBCDIC	See below
Adjusted Rate	6	41	Packed	999999999V99C; dollars
Connect Charge	5	47	"	9999999V99C; dollars
RJE Charge	5	52	"	9999999V99C; dollars

The Charge Suffix indicates that the Total Charge field contains one of the following values:

Charge Suffix	Description
-	Credit amount
+	Debit amount
*	Budget amount
M	Minimum job charge
B	Block time charge
blank	Normal job charge

Each accounting record and all pertinent billing parameters are made available through the parameter list. The user may modify information in the accounting record and calculate the various time and charge fields related to the record.

Upon entry to the user exit routine, the time and charge fields in working storage are initialized and available for calculations. When the record being processed is a summary record from the historical data base, the working storage area will contain the time and charge values accumulated into that record when it was created.

The one-byte EBCDIC disposition indicator, as referenced through the parameter list, is blank upon entry to the user exit routine. A hex 'FF' (high-values) upon entry indicates that an end-of-file condition has occurred and no further record processing is to be performed.

You should take this opportunity to have the exit prepare totals, generate user-formatted reports, close files, and so forth. You can optionally set the one-byte EBCDIC disposition indicator to specify the following action(s) to be taken upon each return to CA JARS:

Disposition Indicator	Description
blank	Default. Normal processing continues with calculating values in working storage area using standard accounting algorithm.
1	Skip standard accounting algorithm calculations. Use values in working storage area as placed there by user exit routine.
2	Reject record from further processing. Read next input accounting record.

A type 1 exit could be used to:

1. Modify the rate tables.
2. Perform unique rate calculations--totally independent of the CA JARS process.
3. Test or change accounting fields.

Exit 2

This exit provides control to the user immediately following completion of the standard accounting algorithm calculations and prior to the construction of user sort control fields, as specified by the SORT statement, for each accounting record being processed. This allows you to perform tests verifying the time and charge calculations or make unique billing adjustments to records reaching this phase of processing. The parameter list and record processing sequence for this exit are identical to that of Exit 1.

Exit 2 is identified in positions 18-25 on the EXITS control statement.

Accounting records and pertinent billing parameters are made available through the parameter list. The user can modify information in the accounting record or any field in the working storage area made available through the parameter list. Upon entry to the user exit routine, the time and charge fields in working storage contain the results of the standard accounting algorithm calculations based on the appropriate RATE, TSORATE, FORMRATE, RJERATE, SNARATE, APPCRATE, and PRIORITY statements.

Alterations in the charge fields must be reflected in the totals, which have already been accumulated at this exit point.

The one-byte EBCDIC disposition indicator, as referenced through the parameter list, is blank upon entry to the user exit routine. A hex 'FF' (high-values) upon entry indicates that an end-of-file condition has occurred and no further record processing is to be performed. The user should take this opportunity to have the exit prepare totals, generate user-formatted reports, close files, and so forth.

The user can optionally set the one-byte disposition indicator to specify the necessary action to be taken upon return to CA JARS. If the disposition indicator has been set by the user to any nonblank character, then the record will be automatically rejected from further processing.

Some possible uses of this exit type are:

- Verifying time or charge calculations.
- Performing unique rate calculations such as a surcharge or a discount.
- Rejecting records from future processing. Remember to reflect these changes in the totals.

Exit 3

This exit provides control to the user during the output phase while generating the user-defined reports. Exit 3 allows users to modify print line information or use summarized data for reporting purposes unique to their installation.

This exit is named in positions 26-33 of the EXITS control statement.

The parameter list for this exit is as follows:

Word	Description
1	Address of Set Code for report currently being generated
2	Address of print line identification code
3	Address of record type indicators
4	Address of Output Data Elements Table

- The Output Data Elements Table is shown in its entirety in the *User Guide.*
- The print line identification code referenced in the parameter list defines the type of print line currently being developed by CA JARS. This code is a one-byte binary number defining the type of information currently stored in the Output Data Elements Table (provided on the next page).

Print Line ID	Description
X'01'	Top line of output data element titles available for display.
X'02'	Bottom line of output data element titles available for display.

Print Line ID	Description
X'03'	Output data elements available for display at detail level (that is, job and job step records).
X'04'	Output data elements available for display at final summarization level.
X'05'	Output data elements available for display at summarization level 1.
X'06'	Output data elements available for display at summarization level 2.
X'07'	Output data elements available for display at summarization level 3.
X'08'	Output data elements available for display at summarization level 4.
X'09'	Output data elements available for display at summarization level 5.
X'50'	Output data elements available for display as level 1 header information.
X'60'	Output data elements available for display as level 2 header information.
X'70'	Output data elements available for display as level 3 header information.
X'80'	Output data elements available for display as level 4 header information.
X'90'	Output data elements available for display as level 5 header information.
X'FF'	End-of-file condition has occurred and no further record processing is to be performed. Previous entry should have been with identification code X'40' for final totals.

If the print line identification code is equal to X'03', then the record type indicator defines which type of record is currently being processed to develop the detail print line. The record type indicator has no meaning when the print line identification code is a value other than X'03'. This indicator is a two-byte EBCDIC field containing one of the following values:

Record Type Indicator	Description
1&blank.	Output Data Elements Table contains job step level information.
&blank.1	Output Data Elements Table contains job level information.
11	Output Data Elements Table contains information for a one-step job.

The Output Data Elements Table referenced in the parameter list contains the output data elements available for display. The *TBL DSP* column defines the displacement, relative to zero, into the table for each output data element. The *LEN* column defines the field length, and the *NOTES* column indicates the field format after editing but prior to printing. The information in this table is available to users for verification and modification purposes. By changing the table information, the user can effectively control the information which will eventually be selected for display based on the DISPLAY statement.

Additionally, this information can be referenced to develop special reports and user records formatted for output by the user exit routine.

Note that only those output data elements specified on a DISPLAY or EDIT statement will contain valid information in the Output Data Elements Table. If the logic of an exit requires access to a data element which is not displayed, an EDIT statement may be used to force editing of the data element and make it available. It is not recommended that numeric operations be performed on numeric output data elements since their formats may change because of dynamic re-editing to avoid high-order digit truncation.

The exit is not passed the actual print line; therefore, you cannot alter the spacing between lines.

If you do not wish to see a specific print line, it can be deleted by returning high values in the print line identification code. Note, however, that the HEADER will still print for each page even if the user deletes all the print lines.

Exits on Installation Tape

There are certain exits which are distributed with the CA JARS system. They are delivered in either source or load module format as appropriate and after SMP APPLY are in the libraries shown in the table below.

Note: Source modules need to be link edited **prior** to execution.

Member Name	Library	Description
SJEX201	CAI.CAJRLOAD	Rate Determination Exit
SPLITJOB	CAI.CAJRSAMP	Split Job Exit
VSPCEXIT	CAI.CAJRSAMP	Virtual Storage Personal Computing (VSPC) Exit
JSIRPGXB	CAI.CAJRLOAD	Special Rundate Exit
CAJRDEXT	CAI.CAJRSAMP	Delimits account code field with commas

Note: JARS/OLF provides alternate chargeback capability online. Using JARS/OLF will eliminate coding and maintaining the rate determination on split job exits. See the *JARS/OLF User Guide* for detailed instructions on establishing rates and distributing cost across several accounting entities.

The Rate Determination Exit (SJEX201)

The function of this Exit 2 exit is to produce reports describing changes made to RATE, TSORATE, FORMRATE, and RJERATE statements, in order to achieve cost recovery goals for the different subsystems reported on by CA JARS, such as TSO, CICS, VM, and Normal/Batch.

An extensive description of this exit is provided in Appendix A of this guide.

The Split Job Exit (SPLITJOB)

A split job exit lets you distribute the proportional costs of a given job among two or more accounting units. It is identified as Exit 2 on the EXITS control statement.

In the sample source code provided, a table is constructed consisting of the following elements:

1. JOBNAME (KEY) corresponding to the name of the job for which costs are to be shared.
2. ACCNTNO (USER INFO) corresponding to the account to which part of the cost is to be charged.
3. PERCENTAGE indicating the portion of the cost for the job that is to be charged to this ACCNTNO.

For each job that is to have its cost split, one CREDIT control statement and multiple DEBIT control statements are produced by the exit. The value in the CREDIT control statement is equal to the sum of the values in the DEBIT control statements for a given job. One DEBIT control statement is generated for each of the account codes for that job based on the corresponding percentages. The account code is put into the first eight (8) positions of the user information field of the DEBIT control statements.

Normally, the exit is called by the first user-defined report in a CA JARS run and the generated CREDIT and DEBIT control statements are applied to the next report. The exit deletes all the records in the first report so that a report will not be produced. This will not affect any subsequent reports.

The report that uses the CREDIT and DEBIT control statements generated by the exit should be sorted by the user information field, causing the distributed charges to accumulate to the various account codes. This sample exit **must** be modified to accommodate your installation's accounting conventions.

This charge distribution capability is useful because:

- The costs of a job can be distributed to the individual accounting groups responsible for its use.
- These split costs can be displayed separately on one report.
- The TOTAL COST for an accounting group can be summarized and a bill produced.
- Utilization statistics for each job are retained even if its cost is split.

When assembling the source code for this exit, CAJRMAC must be concatenated to the SYSLIB DD statement. There is additional documentation in the source code itself.

Note: The split job exit is run with raw SMF data. If you want to use a history file, the source must be changed and reassembled.

The VSPC Exit (VSPCEXIT)

Virtual Storage Personal Computing (VSPC) is an online system designed for users without extensive data processing knowledge; it enables them to manipulate data, text and source programs. It lets you submit jobs for batch processing and retrieve the job output at the VSPC terminal.

There are various SMF records created by VSPC, but this exit deals with only the type 48 (user logoff). It contains the user information, reason for session termination, CPU time, connect time, and DASD I/O count for their terminal session to name just a few fields.

It is up to you to modify and/or maintain this input exit that creates a pseudo type 34 (TSO step termination).

The RPGXA Exit (JSIRPGXA)

This exit, used as an Exit 2, matches *run date* to the day of the week and counts the number of days in the run. Also, it assigns a value to GROUP CODE #3. The *group code* can have the following values and meanings:

```
1 = MONDAY
2 = TUESDAY
3 = WEDNESDAY
4 = THURSDAY
5 = FRIDAY
6 = SATURDAY
7 = SUNDAY
0 = invalid date
```

This exit should be run in conjunction with RPGXC (JSIRPGXC).

The RPGXB Exit (JSIRPGXB)

This exit, used as an Exit 3, matches run date (yymmdd format). Change it to yy/mm/dd, and place the weekday in *blank spaces* (output data element 99).

If the date in the control field is not a valid date, no changes are made. Therefore, at least one of the specified sort control fields should be a valid six-character date field.

The RPGXC Exit (JSIRPGXC)

This exit, when used as an Exit 3, averages intermediate level results over the number of days found at the minor level. The following conditions must be present for this exit to function correctly:

1. Be sure to specify job level detail on the SORT statement (1 in position 50), although detail lines will be suppressed in the report.
2. On the SORT statement, specify ascending sort on Group Code #3 as the major sort field (first sort field on the SORT statement), and specify ascending sort on the *run date* as the minor sort field (third sort on the SORT statement). The other sort fields are open to the user to sort on any field he chooses.
3. Exit 2 must invoke JSIRPGXA which sets the correct group code for day of the week.

Stop time will contain the number of days over which the data has been averaged. The last 8 positions of the description will contain that also.

This exit only calculates averages on the following data elements:

- CPU TIME
- ELAPSED TIME
- RESIDENT TIME
- ACTIVE TIME

- TCB TIME
- SRB TIME
- SERVICE UNITS
- PAGE SECONDS
- TOTAL PAGES
- TAPE COUNT
- DISK COUNT
- LINES PRINTED
- NBR JOB
- NBR STEPS
- TSO JOBS
- SERVICE RATE

Chapter 6: Utilities

The CA JARS File Utility is a generalized utility that lets you process one or more input files in a single execution, and dump, edit, copy, or change data within those files. It is device-independent and can process logical records or blocks of data of any size acceptable to a z/OS or VM system.

The File Utility allows you to selectively dump or copy records from sequential data sets. This selection may be based on either the record's position in the data set, the actual content of a given data field or fields within the record, or both. Processing may begin or end at any sequential block number within a file. The File Utility uses include:

- Sampling Files

  Use the File Utility to display records which meet specified criteria without having to write specialized programs. Varied format files can be input to the File Utility in a single execution.

- Generating Test Files

  Use the File Utility to copy only those records that assure a complete test base of records. Old test files can be modified to check new program functions.

- Copying Data Sets

  Use the File Utility to copy any data set. This provides you with a complete backup of important files. The File Utility automatically provides default file attributes for data sets created by the OUTPUT command. Default output file attributes are determined by the output device type as defined in the *File Attributes Defaults Table* shown later in this chapter. In general, for output to tape the default attributes are:

  ```
  RECFM=VB, LRECL=4092, BLKSIZE=4096
  ```

  When output is directed to a 3380 DASD volume, the default attributes are:

  ```
  RECFM=VB, LRECL=4272, BLKSIZE=4276
  ```

  If you are copying raw SMF data which is normally RECFM=VBS, you must supply the RECFM, LRECL, and BLKSIZE attributes. You can provide these attributes as either DCB information on the DD statement that defines CAIOUT, or as keyword parameters on the File Utility OUTPUT command.

- Testing Programs

  The File Utility provides input and output exit points that can be used to test new programs or new functions.

This section contains the following topics:

CA JARS File Utility Commands Coding Conventions

The File Utility accepts nine command statements. They are:

INPUT=	CHANGE=	EDIT
OUTPUT=	SPECS=	EXIT=
FIND=	DUMP=	HEADER=

Each command, except the EDIT command, has keywords and/or parameters that further qualify the capabilities of the File Utility.

Use the following conventions when coding File Utility Commands:

- With the exception of the EDIT command, all File Utility Commands **must** be coded with an equals sign (=) immediately following the command name.
- All commands must be of a card-image, fixed-format, 80-byte record.
- Commands cannot be continued however, INPUT, FIND and CHANGE commands can be repeated indefinitely.
- Unless otherwise stated, a comma must separate each keyword and/or parameter.
- Comments can be coded on any command, providing that at least one (1) blank separates the last keyword/parameter and the comment.
- All commands can use up to 79 bytes of a statement. Byte 80 is reserved.
- Comments can be entered on separate statements provided that the first nonblank character is an asterisk (*).
- The INPUT, FIND, CHANGE, and SPECS commands are viewed by the File Utility as having a logical relationship. An INPUT command must **precede** FIND, CHANGE, and SPECS commands associated with that INPUT.
- The only required command is INPUT.

INPUT Command

This command is required. It defines the File Utility's input file. Since the File Utility accepts multiple input files, the only limitation on the number of INPUT commands you enter is your storage availability. This command has optional keywords that allow you to qualify the input file attributes. Each INPUT command **must** precede FIND and/or CHANGE commands associated with that particular INPUT command.

```
Command    Operands
INPUT=ddname,START=nn|1,STOP=nn|9999999,SKIP=nn|0,DISP=REJ|SEL,COPY,COUNT
```

where:

ddname

(required) specifies a 1- to 8-character name as found on an OS DD JCL statement or a CMS FILEDEF command.

START=nn|1

(optional) a 1- to 7-digit numeric value that specifies the record (relative to 1) within the input file, to initiate File Utility processing. One (1) is the File Utility default. Zero (0) is invalid.

STOP=nn|9999999

(optional) a 1- to 7-digit numeric value that specifies the number of records (past the start) that terminate processing.

Note: If FIND commands have been entered with a given input, then the STOP= value refers to the records that met the FIND criteria, not the records read from input.

SKIP=nn|0

(optional) a 1- to 7-digit numeric value that instructs the File Utility to process every nn records. Zero (0) is the default.

DISP=REJ|SEL

(optional) SEL is the default disposition signifying that records that meet the associated FIND criteria are selected for further processing. Conversely, REJ signifies that those records be rejected from further processing. The option chosen governs the disposition for all records in the input file.

COPY

(optional) specifies that all records, whether selected or rejected, are to be written to the output file. This option is only valid if you have included an OUTPUT command. If COPY is not specified, then File Utility defaults to writing only selected records to the output file.

COUNT

(optional) specifies that only a count of the records in the input file is to be produced. All record selection/rejection logic and output processing is ignored.

FIND Command

The FIND command is optional. When specified, it must follow the INPUT command that defines the data file to be processed. Its purpose is to define a data string that is used as a search vehicle for selection or rejection of records.

```
Command    Operands
FIND=t'data',POS=nnnnn,ro,OFFSET=nnnnn,OFFSIZ=n,LOGIC=AND|OR,RDW
```

where:

t

(required) is the data type.

c

specifies that the following data is considered to be character data and used as is.

x

specifies that the following data is considered to be hex data.

p

specifies that the following data is considered to be packed data.

data

(required) is the actual data string/target data searched for within an input record. The data must be enclosed in single quotes.

For more information on data specification, see Data Specification Rules in this chapter.

POS=nnnnn

(required) a 1- to 5-digit numeric value specifying the starting position, relative to zero (0), of the target data within the record. For variable length records (RECFM=VB or RECFM=VBS), the standard 4-byte RDW field is ignored for purposes of coding POS=n. That is, POS=0 addresses the first byte of the record itself (not the first byte of the RDW). This rule is adjusted if OFFSET is also coded.

ro

(optional) specifies the relational operand to be used as the comparison condition which the target data must meet. Valid operands are:

- EQ – equal
- NE -- not equal
- LT -- less than
- GT -- greater than
- LE -- less than or equal to
- GE -- greater than or equal to

If omitted, the File Utility defaults to EQ.

OFFSET=nnn

(optional) a 1- to 5-digit numeric value that defines the starting position, relative to zero (0), of a field containing an offset value that when added to the start of the record, addresses a data segment within the record.

When OFFSET= is specified, the keyword POS= refers to the start of the data segment itself, not the start of the record.

OFFSIZ=n

(optional) where n is a value between 1 and 4 that defines the size of the field found at OFFSET=nnnnn.

RDW

(optional) specifies that the value found at OFFSET= (includes a 4-byte length for the record descriptor word).

LOGIC=AND|OR

(optional) The LOGIC= keyword provides you with a means by which FIND commands can be logically connected to form Boolean expressions.

AND

specifies that the next FIND statement must be used in qualifying the data for selection or rejection, and that the data must meet all the criteria specified in each FIND command connected with AND.

OR

specifies that if the target data does not meet the criteria on the current FIND command, the next FIND command is used in qualifying the data for selection or rejection.

If omitted, the File Utility defaults to OR.

CHANGE Command

The CHANGE command is optional. When specified, it must follow the INPUT command that defines the data file to be processed. It is used to define a relative field in an input record to be CHANGEd and the data used to facilitate this. It is not necessary to code FIND command(s) in order to use the CHANGE function. CHANGE commands entered with FIND command(s) are applied only to those records selected by the FIND command(s). Conversely, CHANGE commands entered independently of FIND commands are applied to all records in the input file.

```
    Command     Operands
CHANGE=t'data',POS=nnnnn,OFFSET=nnnnn,OFFSIZ=n,RDW
```

where:

t

(required) is the data type.

c

specifies that the following data is considered to be character data and used as is.

x

specifies that the following data is considered to be hex data.

p

specifies that the following data is considered to be packed.

data

(required) is the actual data string to be placed in the record. The data must be enclosed in single quotes.

POS=nnnnn

(required) a 1- to 5-digit numeric value specifying the starting position, relative to zero (0), of the target data within the record. For variable length records (RECFM=VB or RECFM=VBS), the standard 4-byte RDW field is ignored for purposes of coding POS=n. That is, POS=0 addresses the first byte of the record itself (not the first byte of the RDW). This rule is adjusted if OFFSET is also coded.

OFFSET=nnn

(optional) a 1- to 5-digit numeric value used to defines the starting position, relative to zero (0), of a field containing an offset value that when added to the start of the record, addresses a data segment within the record.

When OFFSET= is specified, the keyword POS= refers to the start of the data segment itself, not the start of the record.

OFFSIZ=n

where n is a value between 1 and 4 that defines the size of the field found at OFFSET=nnnnn.

RDW

(optional) specifies that the value found at OFFSET= includes a 4-byte length for the record descriptor word.

SPECS Command

This command is optional. It gives DOS users a means of defining the specifications of an input file. If it is specified, at least one of the five keywords must be specified. The SPECS command must follow the INPUT command for which the file specifications apply. The SPECS command can be used regardless of your operating system. Note that any values specified override DCB values *found* at open time.

```
   Command       Operands
SPECS=RECFM=fff,LRECL=nnnnn,BLKSIZE=nnnnn,TLBL=11,SYS=nnn|005
```

where:

RECFM=fff

fff is a 1- to 3-character value defining the record format of your input file. Valid operands are:

F

Fixed Format

FB

Fixed Blocked

FBA

Fixed Blocked ASA

V

Variable

VB

Variable Blocked

VBS

Variable Blocked Spanned

U

Undefined

LRECL=nnnnn

nnnnn is a 1- to 5-digit numeric value used to define the logical record length of your input file.

BLKSIZE=nnnnn

nnnnn is a 1- to 5-digit numeric value that defines the block size of your input file.

TLBL=ll

ll is a 2-character value defining the valid TAPE label options of your input file. If omitted, the File Utility defaults to SL. Valid operands are:

SL

Standard Label,Rewind

SR

Standard Label,Rewind,Unload

SN

Standard Label,NoRewind

NL

No label,Rewind

NR

No Label,Rewind,Unload

NN

No Label,NoRewind

SYS=nnn|005

nnn is a 1- to 3-digit numeric value defining the System Logical Unit that your input is assigned to. If omitted, the File Utility defaults to SYS005.

This keyword can be especially helpful to DOS users who are inputting multiple tape files. By assigning your input tape files to different SYS numbers, you can avoid mounting and demounting tapes on the same drive. This option also allows DOS users to have multiple input files residing on different device types.

DUMP Command

The DUMP command is optional. When specified, it can be used to dump an entire data set or only those records that meet the FIND criteria, if specified. The dump command can be modified by one or more parameters:

```
   Command    Operands
DUMP=ON,START=nn|1,STOP=nn|9999999,RESET,FILE=nn
```

where:

ON

(required) specifies that the File Utility is to initiate DUMP processing.

START=nn|1

(optional) a 1- to 7-digit numeric value specifying the starting record in the input file to begin processing. If omitted, the CA JARS File Utility defaults to one (1).

STOP=nn|999999

(optional) a 1- to 7-digit numeric value that specifies the number of records (past the start) that terminates processing.

RESET=

(optional) only valid if STOP= is specified. Indicates to the File Utility that you have multiple input files, and that you want the STOP= value RESET after processing each input file. If omitted, the File Utility terminates DUMP processing only when the STOP= value has been satisfied, which, depending upon the value, could be in any of the input files.

Note: If FIND criteria is specified for a given input file, then both the START= and STOP= values apply to the records that meet the FIND criteria.

FILE=nn

(optional) specifies a 1- to 7-digit numeric value corresponding to the sequence of a specified INPUT file. When specified, this parameter tells the File Utility which input file is to participate in dump processing. The corresponding INPUT file and only that file is DUMPed. If omitted, the File Utility assumes all INPUT files participate in dump processing.

Note: If FILE is specified with RESET, RESET is ignored.

HEADER Command

This command is optional. Only one HEADER statement per run is allowed. The HEADER statement is only valid with the DUMP option. It lets you specify up to 70-bytes of heading information, that is automatically centered by the CA JARS File Utility. If you are inputting multiple files, the File Utility automatically breaks to a new page and print your heading. If only one input file is specified, the File Utility prints your header on the first page of DUMPed output. The heading information must be contained within the single quotes.

```
   Command    Operands
HEADER='.......... heading information ..........'
```

OUTPUT Command

This command is optional. It defines a file to be used as output for the File Utility Program. Only one OUTPUT command per run is allowed. This command has optional keywords that let you qualify the output file attributes.

```
   Command    Operands
OUTPUT=ddname,SKIP=nn|0,RECFM=fff,LRECL=nnnnn,BLKSIZE=nnnnn,TLBL=ll where:
```

ddname

(required) specifies a 1- to 8-character name as found on an OS DD statement, a DOS DLBL/TLBL JCL statement, or a CMS FILDEF command.

SKIP=nn|0

(optional) a 1- to 7-digit numeric value that instructs the File Utility to write every nn records to the output file. Zero (0) is the default.

RECFM=fff

fff is a 1- to 3-character value defining the record format of your output file. Valid operands are:

F

Fixed Format

FB

Fixed Blocked

FBA

Fixed Blocked ASA

V

Variable

VB

Variable Blocked

VBS

Variable Blocked Spanned

U

Undefined

LRECL=nnnnn

nnnnn is a 1- to 5-digit numeric value that defines the Logical Record Length of your output file.

BLKSIZE=nnnnn

nnnnn is a 1- to 5-digit numeric value that defines the Block Size of your output file.

TLBL=ll

(optional) ll is a 2-character value defining the valid TAPE label options of your output file. If omitted, the File Utility defaults to SL. Valid operands are:

SL

Standard Label,Rewind

SR

Standard Label,Rewind,Unload

SN

Standard Label,NoRewind

NL

No label,Rewind

NR

No Label,Rewind,Unload

NN

No Label,NoRewind

EDIT Command

This command is optional. Only one EDIT statement per run is allowed. The EDIT command signals the File Utility to terminate processing after all commands have been verified and EDITed for completeness. No record processing is done. In the non-EDIT mode, the File Utility continues to process even if warning (w-level) messages have been issued. With the EDIT command you can halt the processing and review your command stream for errors. There are no keywords or parameters for this command.

```
   Command    Operands
EDIT
```

EXIT Command

This command is optional. Only one EXIT statement per run is allowed. The EXIT command allows you to enter the name(s) of either an input or output exit or both. The exit name(s) cannot exceed 8 characters in length. There are no keywords for this command.

```
   Command     Operands
EXIT=exit1,exit2 where:
```

exit1

(optional) specifies a 1- to 8-character module name to receive control during File Utility input processing.

exit2

(optional) specifies a 1- to 8-character module name to receive control during File Utility output processing.

To omit the exit1 name and only specify the exit2 name, code the following:

- EXIT=,exit2

To omit the exit2 name and only specify the exit1 name, code the following:

- EXIT=exit1

EXIT Processing

The CA JARS File Utility provides two (2) exit points during record processing. Exits are assumed to be global; that is, if exits are specified, they are called for every input record read and for every output record written. Exit routines must be written in a language that supports standard OS linkage conventions. Exit routines must reside in a library that the File Utility has access to during processing. A BLDL is issued for each exit specified. If it fails, exit processing is bypassed. Exit routines are invoked according to the following standard linkage conventions:

Register	Description
13	Address of a register save area.
14	Return address.
15	Address of the exit routine's entry point.
1	Address of the parameter list.

The first exit point provides control to the user immediately after each input record is read into storage. At this point, you can perform further tests on the record to determine more selection and/or rejection criteria. Separate files can be created from the input data passed to the exit1 routine. Records can also be deleted from input processing.

The parameter list passed to the exit1 routine has the following format:

Word	Description
1	Fullword Address of the input record read.
2	Fullword length of the record.
3	Fullword end-of-file indicator.

The exit1 routine is called for each record in every input file specified. To delete a record from further input processing, register 15, upon return, must contain a value of 4. To request that the exit1 routine not be called again, return register 15 with a value of 8. At end-of-file, the indicator passed to the exit1 routine contains a fullword 4; otherwise it is zero. No record is passed to the exit1 routine with this call. The return value in register 15 from the exit1 routine is also ignored with this call.

The second exit point provides you with control immediately before each record is written to the output file. If the OUTPUT command is not supplied, this exit point is bypassed. If you have coded COPY on any of your INPUT commands, then all records, whether selected or rejected, are written to the output file. It is at this point that you can write your own file of rejected records which can also be used in a future execution of the File Utility. Also, you can selectively let the File Utility create an output file of your choosing. Optionally, you can delete records from further output processing.

The parameter list passed to the exit2 routine is in the following format:

Word	Description
1	Fullword Address of the record.
2	Fullword length of the record.
3	Fullword Selection/Rejection indicator.
4	Fullword final-call indicator.

If a record passes all the criteria tests specified by the FIND command(s), then the Selection/Rejection indicator contains a value of zero (0); otherwise, it contains a value of four (4). If you did not code any FIND commands, then this indicator always contains a zero (0) value. In order to delete a record from output processing, the exit2 routine must return register 15 with a value of 4. To request that the exit2 routine not be called again, return register 15 with a value of 8. At the completion of all input file processing, the final-call indicator contains a value of 4; no record is passed to the exit2 routine with this call. The return value in register 15 from the exit2 routine is also ignored with this call.

Data Specification Rules

The File Utility determines the length of the objective data field from the immediate data entered on a FIND or CHANGE command.

For character data fields, the length assumed is the exact number of characters entered. Padding must be entered as part of the data field itself if it is part of the FIND or CHANGE criteria. The maximum length for character data fields is 32.

For hex data fields, the length assumed is the number of characters entered divided by two (2), except for odd length entries. For example, if you enter a one-byte X'A' as a data field, the File Utility converts this to a one-byte value containing X'0A'. The maximum number of characters that can be entered as hex input is 32, which when converted, is a 16-byte hex field. If an odd number of characters is entered, the File Utility treats them as a one-byte entry. For example, if you enter X'123', it is converted to X'0123' for processing and its length is assumed to be '2'.

For packed data fields, the length assumed is the number of characters entered divided by two (2) + one (1). For example, if you enter a three-byte P'123' as a data field, the File Utility converts this to a two-byte value containing P'123C'. The maximum number of characters that can be entered as packed input is 15, as the CA JARS File Utility supports up to a PL8 data field, and 15 digits is the maximum number (including a sign byte) that fits in a PL8 field.

The only other command that accepts immediate data fields is the HEADER command. This command allows you up to 70 bytes of character data to be displayed on output.

Note: The File Utility does not accept embedded apostrophes in data fields because the it uses apostrophes as delimiters. If specified, it truncates the data when an apostrophe is found, and reports trailing data as an error.

Usage Examples

In examples that specify POS= or OFFSET=, these values are always relative to byte (0) of the data portion of the record not the RDW keyword prefix. The RDW keyword implies that the content at OFFSET presumes that the displacement value includes 4 bytes for a RDW (that is, it is relative to the RDW starting point). The following example illustrates the commands necessary to find and dump all the SMF type 30, subtype 4 records with a jobname of DSTJSOAC:

```
//JOBCARD   JOB
//*
//STEP001 EXEC PGM=JSI,PARM='JSSAU400'
//STEPLIB  DD  DSN=CAI.JARS.CAI.CAJRLOAD,
//         DISP=SHR
//*
//CAIJFPR  DD  SYSOUT=*
//*
//CAIJFSN  DD  SYSOUT=*
//*
//CAIIN    DD  DISP=SHR,
//         DSN=SYSPROG.SMFSAVXA.DATA(-1)
//*
//CAIJFIN  DD  *
*
INPUT=CAIIN
FIND=X'1E',EQ,POS=1,LOGIC=AND
FIND=X'0004',EQ,POS=0,OFFSET=20,OFFSIZ=4,RDW,LOGIC=AND
FIND=C'DSTJSOAC',EQ,POS=0,OFFSET=28,OFFSIZ=4,RDW
DUMP=ON
*
/*
//
```

The following illustrates the commands necessary to find and dump all the SMF type 30, subtype 4 records with a jobname of DSTJSOAC, create an output file, and change the jobname field to DSTJS001:

```
//JOBCARD   JOB
//*
//STEP001 EXEC PGM=JSI,PARM='JSSAU400'
//STEPLIB  DD  DSN=CAI.JARS.CAI.CAJRLOAD,
//         DISP=SHR
//*
//CAIJFPR  DD  SYSOUT=*
//*
//CAIJFSN  DD  SYSOUT=*
//*
//CAIIN    DD  DISP=SHR,
//         DSN=SYSPROG.SMFSAVXA.DATA(-1)
//*
//CAIOUT   DD  DISP=(,CATLG),
//         DSN=CAI.TEST.SMF,UNIT=SYSDA,SPACE=(TRK,(10,10))
//*
//CAIJFIN  DD  *
*
INPUT=CAIIN
FIND=X'1E',EQ,POS=1,LOGIC=AND
FIND=X'0004',EQ,POS=0,OFFSET=20,OFFSIZ=4,RDW,LOGIC=AND
FIND=C'DSTJSOAC',EQ,POS=0,OFFSET=28,OFFSIZ=4,RDW
CHANGE=C'DSTJS001',POS=0,OFFSET=28,OFFSIZ=4,RDW
DUMP=ON
OUTPUT=CAIOUT
*
/*
//
```

In the previous two examples, only those records which have met the FIND criteria are copied to the output file. To have the output file comprised of all the records in the input file, add the COPY keyword to the INPUT command:

INPUT=CAIIN,COPY

The following illustrates the commands necessary for combining two input files to create one output file, and dumping every 50th record in the second input file:

```
//JOBCARD   JOB
//*
//STEP001 EXEC PGM=JSI,PARM='JSSAU400'
//STEPLIB  DD  DSN=CAI.JARS.CAI.CAJRLOAD,
//         DISP=SHR
//*
//CAIJFPR  DD  SYSOUT=*
//*
//CAIJFSN  DD  SYSOUT=*
//*
//INPUT1   DD  DISP=SHR,
//         DSN=VMDATA.DEC92
//*
//INPUT2   DD  DISP=SHR,
//         DSN=VMDATA.JAN93
//*
//VMOUT    DD  DISP=(,KEEP),
//         DSN=CAI.VMDATA.TODATE,UNIT=3400-6,LABEL=(1,SL),
//         VOL=SER=VMTP01
//*
//CAIJFIN  DD  *
*
INPUT=INPUT1
INPUT=INPUT2
DUMP=ON,SKIP=50,FILE=2
OUTPUT=VMOUT
*
/*
//
```

File Attribute Defaults

Since the File Utility is device-independent and automatically sets file attribute default values, this section has been designed to provide you with a comprehensive table of situations and defaults.

File Attributes Defaults Table

The following defaults are set in the following order if they are not in the DCB, DD statement, or LABEL:

Option	Default Value
DSORG	PS

Option	Default Value	
EROPT	ACC	
KEYLEN	0	
RECFM	VB	
BLKSIZE	Optimum BLKSIZE	Device Type
	3625	2311
	4892	2303
	4984	2302
	2000	2321
	4424	2305-1
	3516	2305-2
	3520	2314
	4252	3330 SINGLE DENSITY
	4100	3340
	4628	3350 NATIVE MODE
	4252	3330-1 DUAL DENSITY
	4096	3375
	4096	3390
	4276	3380
LRECL for RECFM=F or U	BLKSIZE	
LRECL for RECFM=V	BLKSIZE-4	

Note: If you try to copy SMF data using the File Utility Output command, you must specify the output file attributes. The File Utility default RECFM is VB; SMF data is RECFM VBS.

File Utility Control Reports

The Control Report consists of two reports:

- File Utility Control Report: synopsis of records processed as directed by the file utility control statements
- Termination Report: summarizes processing by DDname

Samples of these reports follow.

```
CA JARS r12                     CA - JARS    FILE    UTILITY    CONTROL    REPORT                   CAIJRR00  PAGE    2
Resource Accounting SP0                                                                                       14 NOV 1998 11.
INPUT       RECORDS      RECORDS     RECORDS     RECORDS     DELETED     DELETED     INPUT RECS     RECORDS
NAME           READ     SELECTED    REJECTED     CHANGED    BY EXIT1    BY EXIT2       BYPASSED      DUMPED
CAIIN         11003          724       10279           0           0           0              0           0
TOTALS        11003          724       10279           0           0           0              0           0
TOTAL CONTROL CARDS READ = 2.
```

```
CA JARS r12                            T E R M I N A T I O N   R E P O R T                         CAIJFR99   PAGE    1
Resource Accounting SP0                                                                                      14 NOV 1998 11.
** PRODUCT RETURN CODE -- 0000
** MESSAGES -- ** NO MESSAGES PRODUCED **
** FILE USAGE --
   NAME- CAIJFPR                                                               RECORD COUNTS-
                  ACCESS - SAM                                                            0 -INPUT
                  BLKSIZE- 00133                                                         25 -OUTPUT
                  LRECL  - 00133                                                          0 -UPDATED
                  CISIZE - 00000
                  RECFORM- FIXED ANS-PR-CTL
                  USAGE  - OUTPUT MOVE
                  SYS    - LST
                  TAP.OPT- NONE
   NAME- CAIJFIN                                                               RECORD COUNTS-
                  ACCESS - SAM                                                            2 -INPUT
                  BLKSIZE- 00080                                                          0 -OUTPUT
                  LRECL  - 00080                                                          0 -UPDATED
                  CISIZE - 00000
                  RECFORM- FIXED BLOCKED
                  USAGE  - INPUT MOVE
                  SYS    - 000
                  TAP.OPT- NONE
   NAME- CAIIN                                                                 RECORD COUNTS-
                  ACCESS - SAM                                                       11,003 -INPUT
                  BLKSIZE- 04096                                                          0 -OUTPUT
                  LRECL  - 32767                                                          0 -UPDATED
                  CISIZE - 00000
                  RECFORM- VARIABLE BLOCKED SPANNED
                  USAGE  - INPUT LOCATE
                  SYS    - 005
                  TAP.OPT- STANDARD-LABEL REWIND
** START TIME -- 11.03.23 END TIME -- 11.03.33 DURATION -- 00.00.10
```

DOS History File Conversion Utility

The DOS History Conversion Utility program offers the following benefits:

- Provides upward compatibility for the VSE Job Accounting Report System user upgrading to a z/OS environment.
- Provides an organization having diversified processing systems such as VSE, POWER/VS, CICS/VS, RJE, or z/OS with the ability to report on total utilization at the organization level.

These benefits are accomplished by letting you maintain and report on VSE and POWER/VS historical data using CA JARS. The JSITOPS Utility Program translates detail history files (Release 4, Level 7) created as output from the VSE version of this product to detail history files in a format acceptable to CA JARS z/OS. Records with processing IDs of P (POWER/VS), D (DOS Job Accounting), C (CICS), R (RJE), and U (unit record) are converted. All other records are ignored. The translation process allows you to report on utilization and accounting data at the job and job step level.

Note: Do not attempt to produce System Use Reports with the translated data. Several data elements and record types necessary to produce meaningful System Use Reports are not available in this data.

Operating Instructions

The VSE History Conversion Utility Program operates under any OS/VS environment. The following files are required to execute the program:

DDname	Description
STEPLIB	The STEPLIB statement describes the Partitioned Data Set (PDS) that includes JSITOPS and JSITRD4 as members.
CAIJSDOS	The CAIJSDOS statement describes the input history file generated in a previous run of the DOS/VS(E) CA JARS product.
HISTOUT	The HISTOUT statement describes the output history file to be created by CA JARS.
CAIJSPRT	The CAIJSPRT statement describes a SYSOUT data set for messages.
CAIJSNAP	The CAIJSNAP statement describes a SYSOUT data set for CAIJSNAP diagnostics.

The following is sample JCL for the operation of JSITOPS. Installation-dependent parameters are displayed in lowercase letters. **Sample JSITOPS Operations JCL**

```
//JSITOPS  JOB  accounting fields, etc.
//         EXEC PGM=JSITOPS,REGION=1024K
//STEPLIB  DD   library containing JSITOPS load module
//CAIJSPRT DD   SYSOUT=*
//CAIJSNAP DD   SYSOUT=*
//CAIJSDOS DD   DSN=old.format,DISP=OLD,UNIT=tape,
//              DCB=(RECFM=VB,LRECL=2000,BLKSIZE=9000),
//              VOL=SER=mytape,LABEL=(2,BLP)
//HISTOUT  DD   DSN=new.format,DISP=(NEW,KEEP),UNIT=tape,
//              DCB=(RECFM=VB,LRECL=8188,BLKSIZE=8192)
//
```

Control Report

The DOS History Conversion Utility Program prints a Control Report on successful execution.

The Control Report shows total counts of all records processed by the program. The following data elements are shown:

XXX RECORDS WRITTEN

The number of records created from the input data, where XXX defines the type of records written. For example, XXX can be CICS, DOS, RJE, or blank (indicating POWER input).

TOTAL RECORDS WRITTEN

The total number of records written to the HISTOUT file.

Sample Control Report

```
   JSITOPS GENERAL TRANSLATE UTILITY             RUN DATE - 01/03/98
   RELEASE r12                                                PAGE 1
*** JSITOPS00 *** PROCESSING INPUT FILE CAIJSDOS
   *** JSIACCT00 *** HISTOUT FILE COMPLETED.
231    DOS JOB RECORDS WRITTEN
                       844    DOS STEP RECORDS WRITTEN
                     1,436    DOS 1-STEP JOB RECORDS WRITTEN
2,511    TOTAL RECORDS WRITTEN
```

Data Management Utility (DMU)

The Data Management Utility (DMU) provides you with a mechanism to build, update, and merge accounting data for computer-related resources, such as programming, data control, and data entry. This data can then be processed to apply charges, produce reports, and integrate the data into the computer job accounting files.

The Data Management Utility operates by processing input containing utilization and accounting information for computer-related resources, and producing an output file that is used by this product as DOS input. The DMU edits the input and checks for syntax discrepancies that can cause improper processing. Any erroneous data is flagged and a diagnostic message is printed describing the problem. In addition, a database maintenance control report is generated that lists all processing actions taken.

The information the DMU manages includes:

- job or function name
- resource identification
- user identification

- accounting information
- Date
- Utilization
- start time/stop time

The DMU program is written using standard IBM ANS COBOL language. All sorting operations performed by the DMU program are executed using the standard IBM or compatible vendor-supplied SORT/MERGE program via the COBOL SORT verb.

The minimum amount of storage required to execute the DMU program is 1024K bytes plus the amount of core required by the SORT/MERGE program. The maximum amount of storage required varies depending upon the size of the I/O buffers for the SORTWORK data sets.

Note: The Data Management Utility can be found in CAI.CAJRLOAD (SJUTL00).

Input

The Data Management Utility reads input statements that control the functions performed against the DMU job accounting data set. The first statement read is the SETUP statement. All remaining statements define the update action required.

The SETUP statement is required and must be the first statement read by the DMU program. This statement is identified by a blank statement type and action code (positions 1-2).

- The *run date* field is used as the header date on the top of each page of the control report produced by the DMU program. This defaults to the current date.
- The *run type* is a five-character, alphanumeric keyword that defines the type of run requested. The valid keywords and their functions are:
  - UPDAT for modifying old data sets
  - BUILD for creating new data sets
  - MERGE for combining old data sets

  If omitted, the run type defaults to the UPDAT mode.

The following is an example of a typical SETUP statement:

```
position   1    1    2
  1...5....0....5....0
    03/15/94BUPDAT
```

The updated file is written on one output file. The control report header reflects the indicated run date.

SETUP Statement

Position	Field Length	Field Name	Notes
1	1	Statement Type	Must be blank
2	1	Action Code	Must be blank
3-10	8	Run Date	mm/dd/yy or dd/mm/yy; defaults to current
11	1	Reserved	Not used
12-16	5	Run Type	UPDAT for update function BUILD for create function MERGE for merging function
17-80	64	Reserved	Not used

The UPDAT mode indicates that an input Data Management Utility Job Accounting data set to be modified as defined by the input control statements. This mode implies the following processes:

- read, edit, and sort all control statements
- read and sort DMU Job Accounting data set(s)
- write updated DMU Job Accounting data set(s)
- log all functions in Database Maintenance Control Report

The control statements are assumed to be a mixture of ADD, CHANGE and DELETE statements, discussed later in this chapter.

The BUILD mode indicates that the DMU Job Accounting data set is created solely by input control statements. This mode necessarily implies the following processes:

- read, edit, and sort all control statements
- write a new DMU Job Accounting data set
- log all functions in the Database Maintenance Control Report

All control statements are assumed to be ADD statements. No input data set is required, nor is it requested.

The MERGE mode indicates that multiple input DMU job accounting data sets will be merged into one file. This mode implies the following processes:

- read input DMU job accounting data set(s)
- write output job accounting data set(s)

No input control statements (except SETUP statement) are required for this mode. The control report displays the first and last record read from each DMU job accounting data set. A record count for all records read and written is also displayed.

The control statements are used to ADD, CHANGE, or DELETE specific records on the DMU job accounting data sets. These statements are required for the UPDAT and BUILD modes and are not used for the MERGE mode as defined by the run type on the SETUP statement. If present, the control statements can appear in any order since they are automatically sorted to interact with the input files.

The Statement Type is a one-character code defining the functions performed by the control statement.

The Action Code is a one-character code defining the type of record(s) involved in the function.

Control Statement Functions

Statement Type	Action Code	Function
A	S	Add a single-step record
A	J	Add a one-step JOB record
C	S	Change a single-step record
C	J	Change all step records of a single job
D	BLANK	Delete groups of jobs by Job Name
D	S	Delete a single-step record
D	J	Delete all step records of a single job

ADD Statement

The statement type **A** represents an ADD statement. The following data elements, if present, are moved from the input statement to the corresponding fields of the output file record:

- Job Name (Function Name)
- CPU Identification (Resource Identification)
- Run Date
- Partition Identification (User or Subresource Identification)
- Start Time
- Stop Time

- User Information (Accounting Information)
- Cancel Code (Action Code)
- Phase Name (Subfunction Name)
- CPU Time (Utilization)

If the action code is **J,** then a single-step job record is built. An **S** action code assumes that multiple step records are being built for a job. In the latter case, the Job Name, CPU Identification, Run Date, Partition Identification and Start Time must be the same for each step. The Stop Time varies depending on the step Stop Time.

Position	Field Length	Field Name	Notes
1	1	Statement Type	Must be A
2	1	Action Code	Must be J or S
3-10	8	Job Name	
11	1	CPU Identification	
12-19	8	Run Date	mm/dd/yy or dd/mm/yy
20-21	2	Partition ID	
22-27	6	Job Start Time	hhmmss
28-33	6	Step Stop Time	hhmmss
34-49	16	User Information	
50-51	2	Cancel Code	
52-59	8	Phase Name	
60-65	6	CPU Time	99999; Seconds
66-80	15	Reserved	Not used

CHANGE Statement

The statement type **C** represents a CHANGE statement. The following input statement fields must be present to locate the corresponding record(s) on the input tape to be modified:

- CPU Identification (Resource Identification)
- Run Date
- Partition Identification (User or Subresource Identification)
- Job Start Time
- Step Stop Time

In order to modify all Job steps within the same job as defined on the CHANGE statement, the action code should be **J** and the step Stop Time must be coded to further qualify the tape record as the specific and only step within a job to be modified. The following CHANGE statement fields may be used to modify the corresponding input record fields:

- Job Name (Function Name)
- User Information (Accounting Information)
- Cancel Code (Action Code)
- Phase Name (Subfunction Name)
- CPU Time (Utilization)

The update method employed by the DMU program requires the user to code only the fields to be modified. Blank fields on the CHARGE statement indicate that there is no change for that field. Nonblank fields are moved to the corresponding fields in the output tape record. If the user wants an alphanumeric field changed to blanks or a numeric field changed to zeros, then a question mark (?) should be coded in the first position of the appropriate CHANGE statement field. Without this feature it would be impossible to blank out a field on the output tape since all blank fields on a CHANGE statement are ignored. In addition to the previously listed data elements, the Start and Stop Time fields can be modified because they are used to locate the input tape record for modification. The change data or new Start and Stop Time fields are coded in the CHANGE Start Time and CHANGE Stop Time fields. When present, they follow the same rules as the other change fields.

Position	Field Length	Field Name	Notes
1	1	Statement Type	Must be C
2	1	Action Code	Must be J or S
3-10	8	Job Name	
11	1	CPU Identification	
12-19	8	Run Date	mm/dd/yy or dd/mm/yy
20-21	2	Partition ID	
22-27	6	Job Start Time	hhmmss
28-33	6	Step Stop Time	hhmmss
34-49	16	User Information	
50-51	2	Cancel Code	
52-59	8	Phase Name	
60-65	6	CPU Time	99999; Seconds

Position	Field Length	Field Name	Notes
66-71	6	Change Start Time	hhmmss
72-77	6	Change Stop Time	hhmmss
78-80	3	Filler	Not used

DELETE Statement

The statement type **D** represents a DELETE statement. The following input statement fields must be present to locate the corresponding record(s) on the input tape to be deleted:

- CPU Identification (Resource Identification)
- Run Date
- Partition Identification (User or Subresource Identification)
- Job Start Time
- Step Stop Time

In order to delete all job steps within the same job as defined on the DELETE statement, the action code should be **J**; the Step Stop Time is unnecessary. An **S** action code requires the Step Stop Time to further qualify the tape record as the specific and only step within a job for deletion. It should be noted that caution must be used when deleting single steps within a job. Unless the step to be deleted is the last one in a job, you should adjust the Start and Stop Times of the other steps to account for the elapsed time gap that would otherwise occur for the job.

An additional feature is available on the DELETE statement that allows you to delete groups of jobs. You accomplish this by coding a blank action code and the Job Name on the DELETE statement. No other fields are used for this function and there can be as many as fifty DELETE statements. A Job Name table is built for future references. Any input tape record having a matching Job Name in the Job Name Table is deleted.

Position	Field Length	Field Name	Notes
1	1	Statement Type	Must be D
2	1	Action Code	Must be J, S, or blank
3-10	8	Job Name	
11	1	CPU Identification	
12-19	8	Run Date	mm/dd/yy or dd/mm/yy
20-21	2	Partition ID	
22-27	6	Job Start Time	hhmmss

Position	Field Length	Field Name	Notes
28-33	6	Step Stop Time	hhmmss
34-80	47	Reserved	Not used

Output

The Data Management Utility Program's primary output is a VSE job accounting data set. A secondary output is a Database Maintenance Control Report displaying the functions performed during processing.

File Output

The generated VSE job accounting data set is in a format acceptable to the report generator.

Database Maintenance Control Report

The Database Maintenance Control Report is produced for all runs.

For the UPDAT mode, the control report lists all of the DELETE statements with a blank action code. All input tape records containing one of the Job Names listed in this section of the control report have been deleted.

For BUILD and UPDAT modes, the control report lists all of the other control statements in the order they have been processed. The functions indicated by the statements have been completed successfully unless noted by an error indicator and message.

For all input files, the first and last records read are displayed on the control report to verify that all inclusive statement and tape records have been processed. Whenever an input statement or file is closed, tallies of input and output records are printed indicating the number of intermediate and total records processed.

```
               OS JOB ACCOUNTING REPORT SYSTEM   08/24/98
             DATABASE MAINTENANCE CONTROL REPORT      PAGE   1
ACTION CARD IMAGE                 DIAGNOSTIC MESSAGE
SETUP    08/24/98 UPDAT
RECORDS THIS CARD    FILE IN      2
                         OUT      0
CUMULATIVE TOTAL          IN      2
                         OUT      0
```

```
               OS JOB ACCOUNTING REPORT SYSTEM         08/24/98
             DATABASE MAINTENANCE CONTROL REPORT      PAGE   2
ACTION CARD IMAGE                  DIAGNOSTIC MESSAGE
       L05/28/98F3070144J000RB17070150 01 0004B 554HALI10 301$JADUMP
              0000204PC0000100003300052904
       S05/31/98BG125110J00090  125149 05 00090 560ST0010 301ASSEMBLY00  0
              6143R000002000006000003106
RECORDS THIS TAPE   FILE  IN      249
                         OUT        0
CUMULATIVE TOTAL          IN      251
                         OUT        0
```

```
               OS JOB ACCOUNTING REPORT SYSTEM          08/24/98
             DATABASE MAINTENANCE CONTROL REPORT       PAGE   3
ACTION CARD IMAGE                 DIAGNOSTIC MESSAGE
ADD     AJJSITEST  05/27/98BG070505071010TEST JSIUTL     JSIUTL   00001  2
DELETE  DJIEBPTPCH 05/28/98F1072425
RECORDS THIS SORTED FILE  IN       0
                         OUT     249
CUMULATIVE TOTAL          IN     251
                         OUT     249
```

Messages and Codes

The Data Management Utility Program communicates with you via the line printer. All control statement functions and detected user errors are printed in the form on a Database Maintenance Control Report on the printer.

Message	Explanation
INVALID CARD TYPE	The statement type field was not A, C, D, or blank.
INVALID ACTION CODE FOR CARD TYPE	Statement type and action code combination invalid.
TAPE INDICATOR NOT NUMERIC ON SETUP CARD	Must be blank or numeric.
FOUND NO MATCH FOR CHANGE/DELETE CARD	Requested record for CHANGE or DELETE function not found on input tape for UPDAT mode. CHANGE/DELETE statements are invalid for BUILD mode.
INVALID KEYWORD ON SETUP CARD	Must be UPDAT, BUILD or MERGE.
SETUP CARD MISSING	SETUP statement missing or not the first statement read.
DELETE JOB TABLE CAPACITY EXCEEDED	Maximum number of DELETE BY JOB NAME allowed is fifty (50).
DELETE JOB NAME GROUP CONTROL CARD	All input tape records with matching Job Name have been deleted.
CPU TIME FIELDS NOT NUMERIC OR BLANK	Must be blank or numeric.
CHANGE START FIELD NOT NUMERIC OR BLANK	Must be blank or numeric.
CHANGE STOP FIELD NOT NUMERIC OR BLANK	Must be blank or numeric.
START TIME FIELD NOT NUMERIC OR BLANK	Must be blank or numeric.
STOP TIME FIELD NOT NUMERIC OR BLANK	Must be blank or numeric.
INSUFFICIENT SPACE ON WORK01 or WORK02	There is too much data to be held on the named file. See the tallies to determine how many records have been processed.
RECORDS THIS CARD/(TAPE) FILE IN/OUT	Tally of input records from the current statement tape file. Tally of output records generated from the current statement or tape file.
CUMULATIVE TOTAL IN/OUT	Cumulative totals of tallies through the current file.

Operating Instructions

The amount of temporary work area needed on DASD units varies depending upon the anticipated volume of accounting data to be processed and the type of DASD units used.

Job Control Statements

EXEC STATEMENT: to execute the Data Management Utility Program, use the following statement:

```
//STEP1  EXEC  PGM=SJUTL00
```

User Libraries

The STEPLIB and SORTLIB DD statements point to the user's load library and sort library.

```
SORTWK AND WORK DD STATEMENTS
```

The SORTWK01, 02, and 03, and the WORK01 and 02 DD statements are temporary system SORTWORK and DMU program work areas. Define the primary and secondary track allocations to be used in conjunction with the standard procedures for temporary work areas for batch job processing.

```
CAIJSDOS DD STATEMENT
```

The CAIJSDOS DD statement is the DMU job accounting input file.

Tapeout DD Statement

The TAPEOUT DD statement is the updated version of the DOSTAPE.

```
CARDIN DD STATEMENT
```

The CARDIN DD statement describes the Data Management Utility control statements.

Other Considerations

A considerable amount of DASD arm contention occurs during the execution of the Data Management Utility Program because of the number of I/Os performed by the routine. The following points help optimize the DMU program run time:

- Make the assignments for SORTWK01, SORTWK02, and SORTWK03 to different DASD units, if possible, to separate SORTWORK data sets.
- Make the assignment for WORK01 and WORK02 to different DASD units, if possible, to separate the sorted transactions and accounting records data sets.

Run Deck Setups:

The figure below is a sample run deck for executing the Data Management Utility.

```
//STEP1     EXEC PGM=SJUTL00
//SYSOUT    DD   SYSOUT=A
//STEPLIB   DD   DSN=USERLIB,DISP=SHR
//SORTLIB   DD   DSN=SYS1.SORTLIB,DISP=SHR
//SYSUDUMP  DD   SYSOUT=A
//PRINTOUT  DD   SYSOUT=A
//WORK01    DD   DSN=&.&A.,UNIT=SYSDA,SPACE=(TRK,(TT,EE))
//WORK02    DD   DSN=&.&B.,UNIT=SYSDA,SPACE=(TRK,(TT,EE))
//SORTWK01  DD   UNIT=SYSDA,SPACE=(CYL,MM,,CONTIG)
//SORTWK02  DD   UNIT=SYSDA,SPACE=(CYL,MM,,CONTIG)
//SORTWK03  DD   UNIT=SYSDA,SPACE=(CYL,MM,,CONTIG)
//TAPEOUT   DD   DSN=FILENAME,UNIT=,VOL=SER=,DISP=(,KEEP)
//CAIJSDOS  DD   DSN=FILENAME,UNIT=,VOL=SER=,DISP=OLD
//CARDIN    DD   *
      DATA MANAGEMENT UTILITY CONTROL CARDS
/*
```

Chapter 7: Using CA JARS with Special Environments

As an expansion of the CA JARS reporting capabilities, the following interfaces are offered:

- ADABAS Interface
- CICS Interface (CA JARS for CICS)
- Datacom/DB
- DB2 Interface
- IMS Interface
- IDMS Interface (CA JARS IDMS Option)
- Network Accounting Interface
- Roscoe Interface
- Tape Volume Accounting Interface
- VM Interface

All interfaces, with the exception of CA JARS for CICS and the CA JARS IDMS Option Interfaces, are found on the CA JARS distribution tape and supporting documentation (that is, Interface Guides). CA JARS for CICS and CA JARS IDMS Option are available separately.

This chapter is provided to help the systems programmer work more easily with the interfaces. The following pages provide an explanation of how to use Title Tables and an Account Record Matrix, cross-referencing the elements' fields of the account records for all the input data available to the report program.

This section contains the following topics:

Defining Alternate Data Element Title Headings

CA JARS is designed so that a user may make up one or more sets of Output Data Element Title Headings and use them in place of the set provided with the system. This approach eliminates the necessity for large sets of TITLE control statements used to override the default data element titles. The *Title Tables* supplied on the distribution tape are members:

```
JSIRTT01(standard descriptions of output data elements-default)
 JSIRTT02       (titles for CICS descriptions of output data elements)
 JSIRTT03       (titles for APPC descriptions of output data elements)
 JSIRTTMA       (titles for Roscoe descriptions of output data elements)
 JSIRTTQA       (titles for ADABAS descriptions of output data elements)
 JSIRTTVO       (titles for VM descriptions of output data elements)
 JSIRTTMS       (titles for IMS descriptions of output data elements)
 JSIRTTIS       (titles for IDMS descriptions of output data elements.
```

Note: The Title Table for JSIRTTIS is provided on the CA JARS IDMS Option distribution tape.

These tables are in the form of assembly programs which consist of constants and have no executable statements. Each table also contains conditional assembly statements which verify that the tables are the correct length to ensure that titles are not incorrectly modified.

A set of user-specified titles may be defined by completing the following steps:

1. Punch out the appropriate member (JSIRTT01, JSIRTT02, etc.)
2. Change the upper and lower data element title description constants as desired, being careful not to change the length of any description constant (as they must remain the same length as the corresponding output data element).
3. Assemble the modified title table, checking for the presence of MNOTE statements indicating that the table lengths are invalid. Link edit the resultant object deck, specifying a load module name of JSIRTT01.

This sequence of steps causes the distribution set of titles to be replaced with the user-specified set. If several sets of alternate titles are needed or the distribution set is not to be replaced, a number of different load modules may be link edited with phase names of the form JSIRTTxx, where xx is any two characters other than listed above. These alternate title sets may be indicated for a user-defined report by entering the last two positions of the title table load module in positions 10-11 of the DISPLAY control statement.

For example, if a title table is link edited with the name JSIRTT99, then a report with the DISPLAY statement:

```
1DISPLAY 99............
```

uses the titles defined in that load module.

The module JSIRTT01 is the default title table and is used for data element titles if positions 10-11 are blank on the DISPLAY statement. Regardless of which title module is used, title control statements may still be used to override individual titles.

Account Record Matrix

On the following pages is an *expanded* account record matrix. Two interfaces have been added to the matrix, IDMS and Datacom/DB.

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
1	CPU ID	CPU I	CPU ID	CPU ID	57
2-3	Reserved	Reserved	Reserved	Reserved	None
4-9	Rdr Start Date	CICS Start Date	Logon Date	Date Record Written	60
10-15	Rdr Start Time	CICS Start Time	Logon Time	Time Record Written	61
16-23	Jobname/RJE Line Name	Trans/Term ID	User ID	Job Name	04/H1
24-25 1 in 24=Step 1 in 25=Job	Record Ind. 1 in 24=Program Record 2 in 25=Trans. Record	Record Ind. Ind. C'11'	Step Job	Record Ind. C'11'	
26-27	Step Number	Use Count	# of Steps	Trans Count	56
28	Processing ID D,S,T,-, +,*,H,R, or U	Processing ID 'C'	Processing ID ID 'T'	Processing ID 'C'	03
29-36	User Ident.	Term/Trans ID	Account No.	Trans ID	10
37-42	Start Time (HHMMSS)	Trans Start Time (HHMMSS)	Logon Time	Trans Stop Time	12
43-48	Run Date	Trans Start Date	Logon Date	Trans Stop Date	06
49-50	Class/ Partition ID/APPC Class	Trans Type Short/Long/ Pvt.	Reserved	Trans Type	07
51-56	Stop Time	Trans Stop Time	Stop Time	Reserved	13
57-76	Programmer Name/APPC Conversation ID	User Character Field A	Reserved	Reserved	58
77-92	User Info/Acct Field	User Info	Acct Fields	User Info	
93-96	Completion Code/Cancel Cancel	Abend Code	Reserved	Abend Code	09

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
97-98	Termination Indicator	State Indicator (S/M/C)	Reserved	Reserved	59
99-106	Step Name/RJE Remote Name	User Character Field B	Record ID '*VM/370*'	Step Name	11/H0
107-114	Program/ Phase Name/RJE Pswd/APPC TP Name	Program Name	Reserved	PSB Name	05/G9
115-118	Core Allocated	Largest Getmain	Reserved	Reserved	22
119-122	Core Used	Max Core Used	Prod. Value	Reserved	21
123-125	% of Unused Core	Reserved	Reserved	Reserved	23
126-130	Job Number (J/S Level)	Last Task No.	Reserved	Reserved	14
BAT	**ADABAS Interface**	**Roscoe Interface**	**IDMS Interface**	**Datacom/ DB Interface**	**ODE**
1	'S' CPU ID	CPU ID	CPU ID	CPU ID	57
2-3	File ID & Vers Level X'F74C'	Reserved	Reserved	Reserved	None
4-9	Min Start Date YYMMDD (6)	Signon Date	Task Start Date RDR Start Date	Min Start Date	60
10-15	Min Start Time HHMMSS (6)	Signon Time	Task Start Time RDR Start	Min Start Time	61
16-23	Jobname (B)	Jobname 1st 8 chars. of User Signon Key	Logical Term. ID	Job Name	04/H1
24-25	Step Ind '1' (1) Job Ind '1' (1)	Record Ind. C'11	Reserved	Reserved	
26-27	Number of log records included (2)	Reserved	Step Number	Command Count	56
28	Processing ID C (1)	Processing ID T	Processing ID	Processing ID	03
29-36	User ID = '*ADABAS*' (8)	User ID = '*Roscoe*'	Task Code	Reserved	10
37-42	Min Start Time HHMMSS (6)	Signon Time	Task Start Time Step Start Time	Min Start Time	12

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
43-48	Min Start Date YYMMDD (6)	Signon Date	Task Start Date Step Start Date	Min Start Date	06
49-50	Command Code (2)	Job Class 'V'	Trans Type	Job Class	07
51-56	Max End Time HHMMSS (6)	Signoff Time	Trans Stop Time	Max End Time	13
57-76	User/ Terminal ID (5) Followed by '-' Jobname (8) followed by '-' Command Code (2)	User Signon Key	Reserved	Concat of Job Name, Run Unit, Operator ID	58
77-92	User/ Terminal ID (5)	Last 14 chars. User Signon Key	Account Fields (08)	User Account Code	
93-96	Reserved	Reserved	IDMS Indicator	Reserved	09
97-98	Reserved	Reserved	Reserved	Reserved	59
99-106	User/ Terminal ID (5)	VTAM Terminal Name	System ID	Record ID	11/H0
107-114	Command Code (2) Followed by '-' File Number (5)	Roscoe Jobname	Reserved	Program Name	05/G9
115-118	Reserved	Reserved	Stg HWM	Reserved	22
119-122	Reserved	Reserved	Stg HWM	Reserved	21
123-125	Reserved	Reserved	Reserved	Reserved	23
126-130	Reserved	User Session	Task Number	Job Number	14
126	Reserved	Reserved	Reserved	Reserved	
127-128	Number TSO Sessions	Reserved	Reserved	Reserved	B4/H3
129-130	Number Batch Jobs	Reserved	Reserved	Reserved	14
131-132	Priority	Trans Priority Level	Reserved	Reserved	15
133-138	Cards Read	TS Operations	Reserved	Total Message Gets	18
139-144	Lines Printed	File Operations	Lines Printed	Reserved	19

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
145-150	Special Lines Printed	DL/I Operations	Reserved	DL/I Operations	E6
151-156	Cards Punched	TD Operations	Reserved	Total Message Inserts	20
157-160	Prt Forms ID	Relative Line Number	Reserved	Reserved	16
161-164	Punch Forms ID	Operator ID	Reserved	Reserved	17
165-170	Setup/Idle Time	Wait Time	Reserved	Reserved	24
171-176	Elapsed Time	Elapsed Time	Elapsed Time	Elapsed Time	25,65
177-182	CPU Time	Application CPU Time	Total CPU Time	Application CPU Time	26,66
183-188	Ovrhd Time/RJE Active Time/APPC TCB Time	System CPU Time	Reserved	Reserved	27,67/ G7
189-194	Wait Time/RJE Connect Time/APPC SRB Time	Connect Time	Reserved	Reserved	28,68/ G1
195-200	Rdr Queue Time	User Time Field A	Reserved	Reserved	62
201-206	Writer Queue Time	User Time Field B	Reserved	Reserved	63
207-212	Turnaround Time	User Time Field C	Reserved	Reserved	64
213-219	Rdr I/O Cnt/RJE Trans Count/APPC Bytes Sent	Journal Control Count	Spooled Rdr I/O Count	Reserved	33/G2
220-226	Printer I/O Count/RJE NAK Count/APPC Bytes Rec'd	Terminal Msgs.	Spooled Prt I/O Count	Terminal Messages	34/G4
227-233	Punch I/O Count/RJE Data Check Count/APPC # Calls	Terminal I/O Characters	Spooled Punch I/O Count	Reserved	35/G6
234-240	Tape I/O Cnt/RJE Invalid Logon Count/APPC # Conver- sations	User Count Field A	Tape Connect Time	GU/GHU DLI Counts	37/G8
241-247	Disk I/O Cnt/RJE Line Error Count/APPC EXCP Count	User Count Field B	Disk Cyl Blk Hrs	GN/GHN DLI Counts	38/G3
BAT	**ADABAS Interface**	**Roscoe Interface**	**IDMS Interface**	**Datacom/ DB Interface**	**ODE**
126	Reserved	Reserved	Reserved	Reserved	
127-128	Reserved	Reserved	Reserved	Reserved	B4/H3

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
129-130	Reserved	Reserved	Reserved	Reserved	14
131-132	Reserved	Reserved	Reserved	Priority Level	15
133-138	Reserved	Reserved	Cumm Scratch Requests	Reserved	18
139-144	Reserved	Reserved	Cumm Scratch Request	Reserved	19
145-150	Reserved	Reserved	Pages Read	Reserved	E6
151-156	Reserved	Reserved	Pages Written	Reserved	20
157-160	Reserved	Reserved	Reserved	Reserved	16
161-164	Reserved	Reserved	Reserved	Reserved	17
165-170	Reserved	Reserved	Reserved	Reserved	24
171-176	Elapsed Time in Hrs: Dec=5 (6)	Elapsed Time	Elapsed Time	Elapsed Time	25,65
177-182	CPU Time in Hrs Dec=5 (6)	CPU Time	CPU Time	CPU Time	26,66
183-188	Reserved	Reserved	User Time	Reserved	27,67, G7
189-194	Reserved	Reserved	Wait Time	Wait Time	28,68, G1
195-200	Reserved	Reserved	Reserved	Reserved	62
201-206	Reserved	Reserved	Reserved	Reserved	63
207-212	Reserved	Reserved	Reserved	Reserved	64
213-219	Associator I/Os (7)	Reserved	GETSTG Requests	Data EXCPs	33/G2
220-226	Work I/Os (7)	Reserved	FREESTG Requests (NAK) Count	Index EXCPs	34/G4
227-233	Reserved	Reserved	Terminal I/Os	Data I/O	35/G6
234-240	Reserved	Reserved	Pages Requested	Index Logical I/O	37/G8
241-247	Reserved	Disk Accessed I/O	Records	Other EXCPs	38/G3

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
248-254	Other I/O Count/RJE Time Out Count/APPC Device Connect Time (1/1000th sec)	User Count Field C	NonSpooled I/O Count	GNP/GHNP DLI Counts	39/G5
255-261	Total I/O Count	Reserved	Total I/O Count	Total DLI Gets	40
262-265	I/O Index	Reserved	I/O Index	Reserved	78
266	Storage Indicator/ APPC Type	Storage Ind. (R/V/X)	Ind.	Reserved	72
267-273	Page-In Count	Page-In Count	VS Page Reads	Reserved	73
274-280	Page-Out Cnt	Page-Out Cnt	VS Page Writes	Reserved	74
281-287	Total Paging Count	Total Pages	Total VS Pages	Reserved	75
288-291	CPU Paging Rate	CPU Page Rate	CPU Paging Rate	Reserved	77
292-295	Elapsed Paging Rate	Elapsed Page Rate	Elapsed Paging Rate	Reserved	76
296	Group-Code #1	Group-Code #1	Group Code #1	Group-Code #1	84
297	Group-Code #2	Group-Code #2	Group Code #2	Group-Code #2	85
298	Group-Code #3	Group-Code #3	Group Code #3	Group-Code #3	86
299-300	Partition ID	Trans Class	Reserved	Reserved	87
301-308	Input Device Name/APPC Local LU Name	User Character Field C	Project Name	Reserved	A0
309-314	Reader Duration	Reserved	NonTape Dedic. Dev Con Time	Reserved	88
315-320	Writer Duration	Reserved	Reserved	Reserved	89
321-327	Number Swaps	User Count Field D	Reserved	ISRT DLI Counts	A1
328-334	Swap Pages In	User Count Field	Reserved	DLET DLI Counts	A2
335-341	Swap Pages Out	User Count Field F	Reserved	REPL DLI Count	A3
342-348	Total Swap Pages	Reserved	Reserved	Reserved	A4

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
349-350	# Of Readers Used	User Count Field G	Reserved	Reserved	92
351-352	# Of Printers Used	User Count Field H	Reserved	Reserved	93
353-354	# Of Punches Used	User Count Field I	Reserved	Reserved	94
355-356	# Of Tapes Used	User Count Field J	# Tape Mounts	Reserved	95
357-358	# Of Disks Used	User Count Field K	# Disk Mounts	Reserved	96
BAT	**ADABAS Interface**	**Roscoe Interface**	**IDMS Interface**	**Datacom/ DB Interface**	**ODE**
248-254	Data I/Os (7)	Terminal Accesses	Records Current RU	Other Logical I/O	39/G5
255-261	Total I/Os (7)	Total Accesses	Fragments Stored	Total I/O Count	40
262-265	I/O Index	I/O Index	Reserved	Reserved	78
266	Reserved	Reserved	Real CPU Indicator	Reserved	72
267-273	Reserved	Reserved	Number Locks	Reserved	73
274-280	Reserved	Reserved	Number Select Locks	Reserved	74
281-287	Reserved	Reserved	No. Update Locks	Reserved	75
288-291	Reserved	Reserved	Reserved	Reserved	77
292-295	Reserved	Reserved	Reserved	Reserved	76
296	Group Code #1	Group Code #1	Group Code #1	Group Code #1	84
297	Group Code #2	Group Code #2	Group Code #2	Group Code #2	85
298	Group Code #3	Group Code #3	Group Code #3	Group Code #3	86
299-300	Reserved	Reserved	Reserved	Reserved	87
301-308	Reserved	User Terminal ID	Reserved	Reserved	A0

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
309-314	Reserved	Reserved	Reserved	Reserved	88
315-320	Reserved	Reserved	Reserved	Reserved	89
321-327	Reserved	Reserved	Records	Reserved	A1
328-334	Reserved	Reserved	System Serv. Requests	Reserved	A2
335-341	Reserved	Reserved	Reserved	Reserved	A3
342-348	Reserved	Reserved	Reserved	Reserved	A4
349-350	Reserved	Reserved	Reserved	Reserved	92
351-352	Reserved	Reserved	Terminal Errors	Reserved	93
353-354	Reserved	Reserved	Reserved	Reserved	94
355-356	Reserved	Reserved	Reserved	Reserved	95
357-358	Reserved	Reserved	Reserved	Reserved	96
BAT	**CA JARS**	**CICS Interface**	**VM Interface**	**IMS Interface**	**ODE**
359-360	# Of Others Used	User Count Field L	# Other Devices	Reserved	98
361-362	# Of Disks Private	User Count Field M	Disk Mounts	Reserved	97
363-364	# Of Devices Used	Reserved	Reserved	Reserved	32
365-368	TPUTS	Reserved	Statements Punched	Reserved	A5
369-372	TGETS	Reserved	Statements Read	Reserved	A6
373-378	Active Time	Response Time	Virtual CPU Time	Reserved	A7, A8
379-384	Connect Time	Reserved	Connect Time	Reserved	B0, B1
385-390	Allocation Delay Time	Reserved	Reserved	Reserved	B8, B9
391-396	Resident Time	Reserved	Reserved	Reserved	C1, 2
397-402	CPU Time (SRB)	Reserved	CP Overhead	Reserved	C4, 5
403-408	CPU Time (TCB)	Reserved	Virtual CPU Time	Reserved	C7, 8
409-414	Service Units	Reserved	Reserved	Reserved	D1
415-417	Service Rate	Reserved	Reserved	Reserved	D2

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
418-419	CPU Index	Reserved	CPU Index	Reserved	D0
420-425	Page Seconds	Reserved	Reserved	Reserved	D3
426-428	Performance Group	Reserved	Reserved	Reserved	D4
429-431	Input Route Code	Terminal Type & Model	Reserved	Priority	B6
432-434	Print Route Code	Terminal Access Method	Reserved	Region Protect Key	B7
435-436	Job Requeued Indicator	Real CPU Indicator	Reserved	Reserved	B5
437	Day-Of-Week Indicator	Day-Of-Week Indicator	Day-Of-Week Indicator	Reserved	D5
438	Group-Code #4	Group-Code #4	Group Code #4	Reserved	D7
439	Group-Code #5	Group-Code #5	Group Code #5	Reserved	D8
440	Group-Code #6	Group-Code #6	Group Code #6	Reserved	D9
441-442	Forms- Entries	Reserved	Forms Ent.	Reserved	E0
443	Sysout Class	Reserved	Reserved	Reserved	E1
444	Sysout Type	Reserved	Reserved	Reserved	E2
445	Sysout Intv Code	Reserved	Reserved	Reserved	E3
446-453	Output Device Name/APPC Partner LU Name	Reserved	Product Name	Reserved	E4

BAT	ADABAS Interface	Roscoe Interface	IDMS Interface	Datacom/ DB Interface	ODE
359-360	Reserved	Reserved	Reserved	Reserved	98
361-362	Reserved	Reserved	Reserved	Reserved	97
363-364	Reserved	Reserved	Reserved	Reserved	32
365-368	Posted ECBs (4)	Reserved	Reserved	Posted ECBS	A5
369-372	Descriptors Updated (4)	Reserved	Reserved	Descriptors Updated	A6
373-378	Reserved	Active Time	Reserved	Reserved	A7, A8
379-384	Reserved	Connect Time	Reserved	Reserved	B0, B1

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
385-390	Reserved	Reserved	Reserved	Reserved	B8, B9
391-396	Reserved	Resident Time	Reserved	Reserved	C1, C2
397-402	Reserved	Reserved	Reserved	Reserved	C4, C5
403-408	Reserved	CPU Time (TCB)	System Time	Reserved	C7, C8
409-414	Reserved	Reserved	Reserved	Reserved	D1
415-417	Reserved	Reserved	Reserved	Reserved	D2
418-419	CPU Index	Reserved	Reserved	CPU Index	D0
420-425	Reserved	Reserved	Reserved	Reserved	D3
426-428	Reserved	Reserved	Reserved	Reserved	D4
429-431	Reserved	UCB # of Terminal	Reserved	Reserved	B6
432-434	Reserved	Terminal Type	Reserved	Reserved	B7
435-436	Reserved	Reserved	Reserved	Reserved	B5
437	Min Start Day-of-Week Indicator (1)	Day-of-Week Indicator	Day-of-Week Indicator	Day-of-Week Indicator	D5
438	Group Code #4	Group Code #4	Group Code #4	Group Code #4	D7
439	Group Code #5	Group Code #5	Group Code #5	Group Code #5	D8
440	Group Code #6	Group Code #6	Group Code #6	Group Code #6	D9
441-442	Reserved	Reserved	Reserved	Reserved	E0
443	Reserved	Reserved	Reserved	Reserved	E1
444	Reserved	Reserved	Reserved	Reserved	E2
445	Reserved	Reserved	Reserved	Reserved	E3
446-453	Reserved	Roscoe Session ID	Reserved	Reserved	E4
454-459	STD Lines Printed	Reserved	Line Printed Total	Reserved	E5
460-464	STD Pages Printed	Reserved	Reserved	Reserved	E7
465-469	SPCL Pages Printed	Reserved	Reserved	Reserved	E8

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
470-474	Total Pages Printed	Reserved	Reserved	Reserved	E9
475-478	No. Of Tape Mounts	Reserved	# Tape Mounts	Reserved	F0
479-482	No. Of Disk Mounts	Reserved	# Disk Mounts	Reserved	F1
483-486	Absorption Rate	Reserved	Reserved	Reserved	F8
487-502	Acct Field 2	Reserved	2 + 3 Prod.	Reserved	F2
503-518	Acct Field 3	Reserved	Reserved	Reserved	F3
519-534	User Char. Field/APPC User Data Field	User Char. Field	Reserved	Reserved	F4
535-539	User Count Field	User Count Field	Reserved	Reserved	F5
540-544	User Time Field	User Time Field	Reserved	Reserved	F6
545-550	SMF Audit	Reserved	Reserved	Reserved	F9
Fields Beyond This Point May Not Be Referenced By Grouping Logic					
551-556	Processor Time	Processor Time	Processor Time	Processor Time	29, 69
557-562	Processor Charge	Processor Charge	Processor Charge	Processor Charge	44
563-568	I/O Time	Reserved	I/O Time	Reserved	30, 70
569-574	I/O Charge	Reserved	I/O Charge	Reserved	45
575-579	U/R Charge	File I/O Charge	U/R Charge	File I/O Charge	54
580-584	Setup Charge	Reserved	Setup Charge	Reserved	55
585-590	Total Charge/ Credit Amt/Debit Amt/Budget Amt	Total Charge	Total Charge	Total Charge	46, 79, 80, 81
591	Charge Suffix	Charge Suffix	Reserved	Reserved	None
592-597	Adjusted Rate	Adjusted Rate	Adjusted Rate	Reserved	43
598-602	Connect Charge	Reserved	Connect Charge	Reserved	B3
603-607	RJE Charge including APPC	Reserved	Reserved	Reserved	H2
608	Reserved	Reserved	Reserved	Reserved	None
BAT	**ADABAS Interface**	**Roscoe Interface**	**IDMS Interface**	**Datacom/ DB Interface**	**ODE**
454-459	Reserved	Reserved	Reserved	Reserved	E5

BAT	CA JARS	CICS Interface	VM Interface	IMS Interface	ODE
460-464	Reserved	Reserved	Calc Noflow	Reserved	E7
465-469	Reserved	Reserved	Calc Ovflow	Reserved	E8
470-474	Reserved	Reserved	Reserved	Reserved	E9
475-478	Reserved	Reserved	Via Noflow	Reserved	F0
479-482	Reserved	Reserved	Via Ovflow	Reserved	F1
483-486	Reserved	Reserved	Reserved	Reserved	F8
487-502	Reserved	Reserved	Acct Fields 2	Reserved	F2
503-518	Run Cycle (8 positions only)	User Signon Key	Acct Fields 3	Reserved	F3
519-534	Account Code defined in QAACT Table: based on Jobname- Userid Match (16)	Reserved	User Char. Field	Reserved	F4
535-539	Plan Unique Identifier	User Count Field	User Count Field	Reserved	F5
540-544	Delay Time	Reserved	User Time Field	Reserved	F6
545-550	'YNNNNN' (6)	Reserved	Condense Key	Reserved	F9
Fields Beyond This Point May Not Be Referenced By Grouping Logic					
551-556	Processor Time	Processor Time	Processor Time	Processor Time	29, 69
557-562	Processor Charge	Processor Charge	Processor Charge	Processor Charge	44
563-568	I/O Time	I/O Time	Reserved	I/O Time	30, 70
569-574	I/O Charge	I/O Charge	Reserved	I/O Charge	45
575-579	Reserved	Reserved	File I/O Charge	Reserved	54
580-584	Reserved	Setup Charge	Reserved	Reserved	55
585-590	Total Charge	Total Charge	Total Charge	Total Charge	46, 79, 80, 81
591	Charge Suffix	Charge Suffix	Charge Suffix	Reserved	None
592-597	Reserved	Adjusted Rate	Adjusted Rate	Reserved	43
598-602	Reserved	Connect Rate	Reserved	Reserved	B3
603-607	Reserved	Reserved	Reserved	Reserved	H2
608	Reserved	Reserved	Reserved	Reserved	None .fo on

For CA JARS and all Interfaces:

The following output data elements are **not** carried in the Account record, but are either calculated from elements within the Account record or are derived from other sources (specified below):

ODE	Description	Source
01	Control Field	Sort Control
02	Description Field	Description Table
41	% Total Charge Within Job Class	Calculated
42	% Total Charge	Calculated
47	Distributed Charge	Calculated
71	Total Time	Calculated
82	Over/Under Budget	Calculated
83	% Budget Spent	Calculated
90	Elapsed Time - Averaged	Calculated
91	CPU Time - Averaged	Calculated
99	Blank Spaces	Supplied For Editing Print Line
A9	Active Time - Averaged	Calculated
B2	Connect Time - Averaged	Calculated
C0	Allocation Time - Averaged	Calculated
C3	Resident Time - Averaged	Calculated
C6	CPU Time (SRB) - Averaged	Calculated
C9	CPU Time (TCB) - Averaged	Calculated
G0	I/O Count By Device	DEVADDR Table
F7	Control Break Count	Sort Control

Chapter 8: CMF Processing

Processing of CMF (SMF 110) records is accomplished in CA JARS by using a dictionary similar to the standard CICS CMF dictionary that is produced when monitoring is started. Unlike the CICS dictionary, which can be viewed primarily as a table of index elements that provide access to data elements, the CA JARS dictionary is simply a flat-file format, where each record in the file maps an individual element or field in the SMF 110 record. This construct was designed to simplify maintenance and to allow manual user modification in extreme circumstances where the fixed architecture design doesn't satisfy the businesses accounting requirements.

The layout of the dictionary is presented in the section CA JARS CMF Dictionary Record Layout in this chapter. For the most part it follows the format of the DFHMCTDR macro with the addition of an EXTDATA offset that tells CA JARS where to place the field in the EXTDATA record. Multiple dictionaries may be concatenated on the CAIJSDCI ddname to support various releases of CICS in the same CA JARS run.

This section contains the following topics:

CMF Processing Flow

Follow the steps below to build a dictionary and process CMF records.

1. Build the dictionary using either CAJRJCL member CMFUTIL1 (CICS 1.7 through 2.1) or CMFUTIL2 (CICS 3.2 and above). Refer to the Building the Dictionary section of this chapter for details on CMFUTIL control statements.

2. Process the SMF CMF records into CA JARS:

   - Define the CICS CMF dictionary generated above to the JCL using the CAIJSDCI ddname statement. See the chapter "Operating Instructions" in the *User Guide* for details.

   - Select the CMF option, column 38, of the EXTDATA statement by indicating a 'Y'. Refer to the EXTDATA control statement in the Control Statement and Tables chapter of the *User Guide*.

   - Retain the account file or generate a level 7 history file (SORT statement) job for subsequent EXT reporting using either CA Earl or CA Easytrieve.

3. Refer to the EXTDATA Reporting chapter of the *User Guide* for details on generating CMF EXT Reports.

The EXTDATA Record

The EXTDATA record for CMF data is a superset record containing the appropriate data for all supported releases of CICS along with an area set aside for future expansion. There is also a fixed area to be used by installations to support inclusion of user data that is collected by user event monitoring points (EMPs). The "fixed" user area has been designed to accommodate various monitoring scenarios that an installation might choose. Because the EXTDATA record matches the IRD record format in JARS/OLF and CA PMA Chargeback, it is highly recommended that CA PMA Chargeback users do not modify the format of the IRD or ORD for the CMF record (JARS/OLF users do not have this capability).

A single character user field definition in an EMP may actually be a composition of one or more data fields, thereby designating a group level field. CA PMA Chargeback users have the option in this case to redefine the ORD fixed user field that the data is moved to, for purposes of referencing the individual fields within the group level field. The following fixed user fields are defined in the EXTDATA record:

EXTDATA Name	IRD/ORD Name	Len	Description
EXTC1L04	C1L04	CL4	4 byte character user area
EXTC2L04	C2L04	CL4	4 byte character user area
EXTC3L08	C3L08	CL8	8 byte character user area
EXTC4L08	C4L08	CL8	8 byte character user area
EXTC5L16	C5L16	CL16	16 byte character user area
EXTC6L16	C6L16	CL16	16 byte character user area
EXTC7L50	C7L50	CL50	50 byte character user area
EXTP1	P1	PL8	8 byte packed user area
EXTP2	P2	PL8	8 byte packed user area
EXTP3	P3	PL8	8 byte packed user area
EXTP4	P4	PL8	8 byte packed user area

Building the Dictionary

The sample JCL data set, CAJRJCL, contains two jobs that have been provided to automate the process of creating the CMF dictionary; CMFUTIL1 and CMFUTIL2. Under normal conditions these utilities need only be run once for each unique CICS release that is installed. If your installation uses different Monitoring Control Tables (MCTs) that specify different EMP's for the performance class, the utilities may need to be run against every CICS region that has a unique MCT. This depends on whether or not you choose to write out user data fields to the EXTDATA record, and how these fields differ from one MCT to the next. Both jobs execute using a series of control statements, required and optional. Control statements consist of a KEYWORD followed by one or more operands. The following rules concern the syntax of the keywords:

- Keywords can start anywhere but must end before column 72.
- An "*" in column 1 denotes a comment.
- A non-blank character in column 72 denotes a continuation.
- An operand must complete on the same line it was started on. Continuation of a broken operand is not permitted.
- One or more blanks may be used in place of delimiters shown, if desired.

CMFUTIL1 - Create Dictionary for CICS 1.7 through 2.1.2

CMFUTIL1 is only used for CICS releases 1.7 through 2.1.2. If your site is only running CICS 3.2 and above, then this job does not apply to you and you can proceed to the information pertaining to the CMFUTIL2 utility.

The CMFUTIL1 utility contains two steps. The first step creates an SMF format data set containing the CICS CMF dictionary record. The record is built using both predefined CICS fields and user-defined fields that are extracted from the MCT for the respective CICS region. The dictionary record is written to a data set associated with the CAJRCMFO ddname. This utility uses the following control statements:

SYSID=sysid

Identifies the system that the SMF data sets belong to

APPLID=applid

The VTAM APPLID of the CICS region that you want this dictionary built for

MCT=xx

The suffix for the monitoring control table (MCT) that was specified at CICS startup for the CICS region that this dictionary is being built for.

Step two builds the dictionary using the CMF dictionary created in the first step. The output dictionary data set from step one is identified as input to this step using the CAJRCMFI ddname. The dictionary that is created in this step is specified on the CAJRCMFO ddname. The output of this step is used as input to the JSIMAIN program using the CAIJSDCI DD statement. The program that is executed by this step is the same as step two of the CMFUTIL2 job.

CMFUTIL2 - Create Dictionary for CICS 3.2 and above

The CMFUTIL2 utility contains two steps. The first step executes the IBM DFHMNDUP utility to build the CICS CMF dictionary record. The record is built using both predefined CICS fields and user-defined fields that are extracted from the MCT for the respective CICS region. Please refer to the *CICS/ESA Operations Guide* for a description of the control statements used with this utility.

Step two executes program CAJ1CMU3 to build the dictionary using the CMF dictionary created in the first step. The output dictionary data set from step one is identified as input to this step using the CAJRCMFI ddname. The dictionary that is created in this step is specified on the CAJRCMFO ddname. The output of this step is used as input to the JSIMAIN program using the CAIJSDCI DD statement. This utility uses the following control statements, all of which are optional:

SYSID=ALL/YES

Specifies whether or not the dictionary being built is exclusive to the SYSID contained in the input data set. If "YES" is specified, then all records in the output dictionary contain the SYSID of the input dictionary. The default is "ALL" if not specified.

APPLID=ALL/YES

Specifies whether or not the dictionary being built is exclusive to the APPLID contained in the input data set. If "YES" is specified, then all records in the output dictionary contain the APPLID of the input dictionary. The default is "ALL" if not specified.

USERFLD=aaaa,bbbbbbbb,aaaa,bbbbbbbb,aaaa,bbbbbbbb....

Specifies the user fields that are to be written from the SMF performance monitoring records to the EXTDATA record during JSIMAIN processing. Operand aaaa identifies the user defined field name in the IRD/ORD. Operand bbbbbbbb identifies the informal name of the user defined field that was specified in the DFHMCT TYPE=EMP Count, Clock, or Field operands for the performance monitoring class. When selecting a user defined IRD/ORD field name, make sure that the field that you have chosen is the correct data type for the field that is being moved. For example, counter and clock fields must use P1 through P4. Numeric fields that are smaller than eight bytes are right justified with leading zeros. Character fields that are moved to a larger field then the size of the data being moved is left justified and padded with blanks. When selecting a character field, try to choose the best fit possible, saving larger fixed fields for additional data.

CA JARS CMF Dictionary Record Layout

The following table describes the layout of the Dictionary entry. These records are produced from the CMFUTIL1 and CMFUTIL2 jobs as detailed on the previous page.

OFFSET	FIELD	LEN	DESCRIPTION
(REL TO 1)			
000001	DCTSYS	CL4	SYSID THIS DICT ('ALL' FOR GLOBAL)
000006	DCTAPPL	CL8	APPLID THIS DICT ('ALL' FOR GLOBAL)
000015	DCTREL	CL3	RELEASE INDICATOR (212, 330 ETC)
000019	DCTCLASS	CL8	CLASS NAME (DFHTASK, DFHTERM ETC)
000028	DCTTYPE	CL1	DATA TYPE
000030	DCTIDENT	CL3	FLD ID (REMAINS SAME ACROSS REL'S)
000034	DCTLEN	CL4	LENGTH OF ENTRY
000041	DCTCON	CL4	CONNECTOR (CHANGES ACROSS REL'S) (HEX)
000047	DCTHDR	CL8	FIELD TITLE
000058	DCTOFFS	CL4	FIELD OFFSET (ZEROES FOR 2.1.2)
000066	DCTEXTOF	CL4	OFFSET INTO EXTDATA RECORD (HEX)

Example:

```
column

....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
   ALL  ALL      212 DFHTERM  S 100 0008   0059  IRIOWTT    0000    01EA
```

This entry applies to all sysids, all applids, and release 2.1.2 of CICS. The class is DFHTERM, type is 'S', the field id (CMF) is 100, the length is 8, the connector is x'59', the field title is 'IRIOWTT', and the output offset in the EXTDATA record is x'1EA'.

CA JARS CMF EXTDATA Record Layout

```
***********************************************************************
*    (6) THE EXT DATA RECORD LAYOUT TO BE WRITTEN OUT
EXTDATA  DSECT  ,
 *
*        FIRST PART OF RECORD IS EXT DATA HEADER FOR EARL/XBACK
*
EXTRDW   DS    XL4                    RDW FOR EXT DATA RECORD
EXTCPUID DS    CL1                    CPUID FROM SYSID
EXTLEVEL DC    CL2'A@'                LEVEL INDICATOR MUST BE 'A@'
EXTDATE  DS    CL6                    DATE = YYMMDD
EXTTIME  DS    CL6                    TIME = HHMMSS
EXTHTYPE DC    CL4'CMCC'              CONSTANT 'CMCC'
EXTSSID  DS    CL4                    SUBSYS ID
EXTIND1  DC    CL2'11'                1ST IND SET TO '11'
EXTIND2  DC    XL2'0000'              2ND IND SET TO '0000'
EXTIND3  DC    CL1':'                 3RD IND SET TO ':'
*
*        START MAJOR SORT KEY, DATA FROM SMF HEADER AND PRODUCT
*        SECTIONS, USED TO SEPERATE BY SYSTEM/CICS REGION AND RECORD
*        SUBTYPE (CURRENTLY ONLY PERFORMANCE RECORDS SUPPORTED)
 *
EXTMNSID DS    XL4                    SYSID
EXTMNSSI DS    CL4                    CONSTANT 'CICS'
EXTMNPRN DS    CL8                    PRODUCT NAME (APPLID)
 EXTMNSPN DS    CL8                    SPECIFIC (APPLID)
 EXTMNSTY DS    XL2                    RECORD SUBTYPE
*
*        DATA RECORD FOLLOWS, A "Y" BELOW THE RELEASE LEVELS AT RIGHT
*        INDICATE WHICH FIELDS CONTAIN DATA FOR WHICH RELEASES. AN "N"
*        INDICATES THE FIELD IS NOT PRESENT IN A PARTICULAR RELEASE.
 *        ALSO IN PARENTHESIS AFTER THE FIELD DESCRIPTION IS THE CICS
*        FIELD ID (FLDID) WHICH REMAINS CONSTANT ACROSS RELEASES OF
*        CICS AND SHOWS THE SOURCE OF EACH EXT DATA FIELD.
 *
*
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
*
*
EXTTRAN  DS    CL4    X'0036'    TRAN          (001)  Y Y Y Y Y Y Y Y
EXTTERM  DS    CL4    X'003A'    TERM          (002)  Y Y Y Y Y Y Y Y
EXTUSER  DS    CL8    X'003E'    USERID        (089)  Y Y Y Y Y Y Y Y
EXTTYPE  DS    CL4    X'0046'    TTYPE         (004)  Y Y Y Y Y Y Y Y
EXTSTRTD DS    PL4    X'004A'    START DATE    (005)  Y Y Y Y Y Y Y Y
EXTSTRTT DS    PL4               START TIME    (005)  Y Y Y Y Y Y Y Y
EXTSTOPD DS    PL4    X'0052'    STOP DATE     (006)  Y Y Y Y Y Y Y Y
EXTSTOPT DS    PL4               STOP TIME     (006)  Y Y Y Y Y Y Y Y
```

```
EXTTRNUM DS    PL4    X'005A'    TRANNUM       (031)  Y Y Y Y Y Y Y Y
EXTTRPRI DS    XL4    X'005E'    TRANPRI       (109)  N Y Y Y Y Y Y Y
EXTTCLAS DS    CL8    X'0062'    TCLSNAME (166)(110)  N Y Y Y Y Y Y Y
EXTLUNME DS    CL8    X'006A'    LUNAME        (111)  N Y Y Y Y Y Y Y
EXTPGMNM DS    CL8    X'0072'    PGMNAME       (071)  Y Y Y Y Y Y Y Y
EXTNETNM DS    CL20   X'007A'    NETNAME       (097)  N Y Y Y Y Y Y Y
EXTUOWID DS    CL8    X'008E'    UOWID         (098)  N Y Y Y Y Y Y Y
EXTRSYS  DS    CL4    X'0096'    RSYSID        (130)  N N Y Y Y Y Y Y
EXTTSKFL DS    CL4    X'009A'    TASKFLAG      (064)  Y Y Y Y Y Y Y Y
EXTABCO  DS    CL4    X'009E'    ABCODEO  (099)(113)  Y Y Y Y Y Y Y Y
EXTABCC  DS    CL4    X'00A2'    ABCODEC       (114)  Y Y Y Y Y Y Y Y
EXTRTYP  DS    CL4    X'00A6'    RTYPE         (112)  N Y Y Y Y Y Y Y
EXTMSGI1 DS    XL4    X'00AA'    TCMSGIN1      (034)  Y Y Y Y Y Y Y Y
EXTCHRI1 DS    XL4    X'00AE'    TCCHRIN1      (083)  Y Y Y Y Y Y Y Y
EXTMSGO1 DS    XL4    X'00B2'    TCMSGOU1      (035)  Y Y Y Y Y Y Y Y
EXTCHRO1 DS    XL4    X'00B6'    TCCHROU1      (084)  Y Y Y Y Y Y Y Y
EXTMSGI2 DS    XL4    X'00BA'    TCMSGIN2      (067)  Y Y Y Y Y Y Y Y
EXTCHRI2 DS    XL4    X'00BE'    TCCHRIN2      (085)  Y Y Y Y Y Y Y Y
EXTMSGO2 DS    XL4    X'00C2'    TCMSGOU2      (068)  Y Y Y Y Y Y Y Y
EXTCHRO2 DS    XL4    X'00C6'    TCCHROU2      (086)  Y Y Y Y Y Y Y Y
EXTM62I2 DS    XL4    X'00CA'    TCM62IN2      (135)  N N Y Y Y Y Y Y
EXTC62I2 DS    XL4    X'00CE'    TCC62IN2      (137)  N N Y Y Y Y Y Y
EXTM62O2 DS    XL4    X'00D2'    TCM62OU2      (136)  N N Y Y Y Y Y Y
EXTTCALC DS    XL4    X'00D6'    TCALLOCT      (069)  Y Y Y Y Y Y Y Y
EXTSCUC1 DS    XL4    X'00DA'    SCUGETCT      (054)  Y Y Y Y Y Y Y Y
EXTSCUC2 DS    XL4    X'00DE'    SCUGETCT      (105)  N Y Y Y Y Y Y Y
EXTSCCC1 DS    XL4    X'00E2'    SCCGETCT      (117)  N Y Y Y Y Y Y Y
EXTSCCC2 DS    XL4    X'00E6'    SCCGETCT      (120)  N Y Y Y Y Y Y Y
EXTSUHW1 DS    XL4    X'00EA'    SCUSRHWM      (033)  Y Y Y Y Y Y Y Y
EXTSUHW2 DS    XL4    X'00EE'    SCUSRHWM      (106)  N Y Y Y Y Y Y Y
EXTSC2HW DS    XL4    X'00F2'    SC24CHWM      (116)  N Y Y Y Y Y Y Y
EXTSC3HW DS    XL4    X'00F6'    SC31CHWM      (119)  N Y Y Y Y Y Y Y
EXTSCUR1 DS    PL8    X'00FA'    SCUSRSTG      (095)  Y Y Y Y Y Y Y Y
EXTSCUR2 DS    PL8    X'0102'    SCUSRSTG      (107)  N Y Y Y Y Y Y Y
EXTSC24C DS    PL8    X'010A'    SC24COCC      (118)  N Y Y Y Y Y Y Y
EXTSC31C DS    PL8    X'0112'    SC31COCC      (121)  N Y Y Y Y Y Y Y
EXTSTGHW DS    XL4    X'011A'    PCSTGHWM      (087)  Y Y Y Y Y Y Y Y
EXT31AHW DS    XL4    X'011E'    PC31AHWM      (139)  N Y Y Y Y Y Y Y
EXT24BHW DS    XL4    X'0122'    PC24BHWM      (108)  N Y Y Y Y Y Y Y
EXT31CHW DS    XL4    X'0126'    PC31CHWM      (142)  N Y Y Y Y Y Y Y
EXT24CHW DS    XL4    X'012A'    PC24CHWM      (143)  N Y Y Y Y Y Y Y
EXT31RHW DS    XL4    X'012E'    PC31RHWM      (122)  N Y Y Y Y Y Y Y
EXT24RHW DS    XL4    X'0132'    PC24RHWM      (162)  N N Y Y Y Y Y Y
EXT31SHW DS    XL4    X'0136'    PC31SHWM      (161)  N N Y Y Y Y Y Y
EXT24SHW DS    XL4    X'013A'    PC24SHWM      (160)  N N Y Y Y Y Y Y
EXTFCGET DS    XL4    X'013E'    FCGETCT       (036)  Y Y Y Y Y Y Y Y
EXTFCPUT DS    XL4    X'0142'    FCPUTCT       (037)  Y Y Y Y Y Y Y Y
EXTFCBRW DS    XL4    X'0146'    FCBRWCT       (038)  Y Y Y Y Y Y Y Y
EXTFCADD DS    XL4    X'014A'    FCADDCT       (039)  Y Y Y Y Y Y Y Y
```

```
EXTFCDEL DS    XL4    X'014E'    FCDELCT       (040)  Y Y Y Y Y Y Y Y
EXTFCTOT DS    XL4    X'0152'    FCTOTCT       (093)  Y Y Y Y Y Y Y Y
EXTFCMCT DS    XL4    X'0156'    FCAMCT        (070)  Y Y Y Y Y Y Y Y
EXTDGET  DS    XL4    X'015A'    TDGETCT       (041)  Y Y Y Y Y Y Y Y
EXTDPUT  DS    XL4    X'015E'    TDPUTCT       (042)  Y Y Y Y Y Y Y Y
EXTDPUR  DS    XL4    X'0162'    TDPURCT       (043)  Y Y Y Y Y Y Y Y
EXTDTOT  DS    XL4    X'0166'    TDTOTCT       (091)  Y Y Y Y Y Y Y Y
EXTTSGET DS    XL4    X'016A'    TSGETCT       (044)  Y Y Y Y Y Y Y Y
EXTTSPTA DS    XL4    X'016E'    TSPUTACT      (046)  Y Y Y Y Y Y Y Y
EXTTSPTM DS    XL4    X'0172'    TSPUTMCT      (047)  Y Y Y Y Y Y Y Y
EXTTSTOT DS    XL4    X'0176'    TSTOTCT       (092)  Y Y Y Y Y Y Y Y
EXTBMSMP DS    XL4    X'017A'    BMSMAPCT      (050)  Y Y Y Y Y Y Y Y
EXTBMSIN DS    XL4    X'017E'    BMSINCT       (051)  Y Y Y Y Y Y Y Y
EXTBMSOT DS    XL4    X'0182'    BMSOUTCT      (052)  Y Y Y Y Y Y Y Y
EXTBMSTL DS    XL4    X'0186'    BMSTOTCT      (090)  Y Y Y Y Y Y Y Y
EXTPCLNK DS    XL4    X'018A'    PCLINKCT      (055)  Y Y Y Y Y Y Y Y
EXTPCXCT DS    XL4    X'018E'    PCXCTLCT      (056)  Y Y Y Y Y Y Y Y
EXTPCLOD DS    XL4    X'0192'    PCLOADCT      (057)  Y Y Y Y Y Y Y Y
EXTJCPUW DS    XL4    X'0196'    JCPUWRCT      (058)  Y Y Y Y Y Y Y Y
EXTICPUI DS    XL4    X'019A'    ICPUINCT      (059)  Y Y Y Y Y Y Y Y
EXTSYNCT DS    XL4    X'019E'    SPSYNCCT      (060)  Y Y Y Y Y Y Y Y
EXTSZALC DS    XL4    X'01A2'    SZALLOCT      (150)  N N Y Y Y Y Y Y
EXTSZRCV DS    XL4    X'01A6'    SZRCVCT       (151)  N N Y Y Y Y Y Y
EXTSZEND DS    XL4    X'01AA'    SZSENDCT      (152)  N N Y Y Y Y Y Y
EXTSZTRT DS    XL4    X'01AE'    SZSTRTCT      (153)  N N Y Y Y Y Y Y
EXTSZCHO DS    XL4    X'01B2'    SZCHROUT      (154)  N N Y Y Y Y Y Y
EXTSZCHI DS    XL4    X'01B6'    SZCHRIN       (155)  N N Y Y Y Y Y Y
EXTSZALL DS    XL4    X'01BA'    SZALLCTO      (157)  N N Y Y YJSJSA002
EXTSZCVT DS    XL4    X'01BE'    SZRCVTO       (158)  N N Y Y Y Y Y Y
EXTSZTOT DS    XL4    X'01C2'    SZTOTCT       (159)  N N Y Y Y Y Y Y
*
*        FOLLOWING FIELDS ARE ALL FROM 'STOPWATCH' TYPE FIELDS
*
EXTUSRDS DS    XL4    X'01C6'    USRDISPT      (007)  Y Y Y Y Y Y Y Y
EXTUSRPT DS    XL4    X'01CA'    USRCPUT       (008)  Y Y Y Y Y Y Y Y
EXTSUSTM DS    XL4    X'01CE'    SUSPTIME      (014)  Y Y Y Y Y Y Y Y
EXTDSPWT DS    XL4    X'01D2'    DISPWTT       (102)  N Y Y Y Y Y Y Y
EXTWTTME DS    XL4    X'01D6'    EXWTTIME      (103)  N Y Y Y Y Y Y Y
EXTTCIWT DS    XL4    X'01DA'    TCIOWTT       (009)  Y Y Y Y Y Y Y Y
EXTFCIWT DS    XL4    X'01DE'    FCIOWTT       (063)  Y Y Y Y Y Y Y Y
EXTJCIWT DS    XL4    X'01E2'    JCIOWTT       (010)  Y Y Y Y Y Y Y Y
EXTSIWT  DS    XL4    X'01E6'    TSIOWTT       (011)  Y Y Y Y Y Y Y Y
EXTIRIWT DS    XL4    X'01EA'    IRIOWTT       (100)  Y Y Y Y Y Y Y Y
EXTTDIWT DS    XL4    X'01EE'    TDIOWTT       (101)  N Y Y Y Y Y Y Y
EXTPCLTM DS    XL4    X'01F2'    PCLOADTM      (115)  N Y Y Y Y Y Y Y
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
```

```
EXTSPDLY DS    XL4    X'01F6'    DSPDELAY      (125)  N N Y Y Y Y Y Y
EXTCLDLY DS    XL4    X'01FA'    TCLDELAY      (126)  N N Y Y Y Y Y Y
EXTXTDLY DS    XL4    X'01FE'    MXTDELAY      (127)  N N Y Y Y Y Y Y
EXTNQDLY DS    XL4    X'0202'    ENQDELAY      (129)  N N Y Y Y Y Y Y
EXTU61WT DS    XL4    X'0206'    LU61WTT       (133)  N N Y Y Y Y Y Y
EXTU62WT DS    XL4    X'020A'    LU62WTT       (134)  N N Y Y Y Y Y Y
EXTSZWWT DS    XL4    X'020E'    SZWAIT        (156)  N N Y Y Y Y Y Y
EXTRMITM DS    XL4    X'0212'    RMITIME       (170)  N N Y Y Y Y Y Y
EXTMISSP DS    XL4    X'0216'    RMISUSP       (171)  N N Y Y Y Y Y Y
EXTCEDLY DS    XL4    X'021A'    DCEDELAY      (175)  N N Y N N N N N
EXTCESWT DS    XL4    X'021E'    DCESWAIT      (176)  N N Y N N N N N
*
*        END OF 'STOPWATCH TYPE FIELDS'
*
EXTOPER  DS    CL4    X'0222'    OPR           (003)  Y Y Y N N N N N
EXTTCSTG DS    XL4    X'0226'    TCSTG         (104)  N Y N N N N N N
EXTP31HW DS    XL4    X'022A'    PC31UHWM      (140)  N Y N N N N N N
EXTP24HW DS    XL4    X'022E'    PC24UHWM      (141)  N Y N N N N N N
EXTPGICT DS    XL4    X'0232'    PAGINCT       (061)  Y N N N N N N N
EXTNEXCT DS    XL4    X'0236'    MNEXCCT       (032)  Y N N N N N N N
EXTC62O2 DS    XL4    X'023A'    TCC62OU2      (138)  N N Y Y Y Y Y Y
EXTSTATL EQU   *-EXTDATA
*
*        FIXED FORMAT AREA FOR USER DEFINED FIELDS
*
EXTC1L04 DS    CL4    X'023E'
EXTC2L04 DS    CL4    X'0242'
EXTC3L08 DS    CL8    X'0246'
EXTC4L08 DS    CL8    X'024E'
EXTC5L16 DS    CL16   X'0256'
EXTC6L16 DS    CL16   X'0266'
EXTC7L50 DS    CL50   X'0276'
EXTP1    DS    PL8    X'02A8'
EXTP2    DS    PL8    X'02B0'
EXTP3    DS    PL8    X'02B8'
EXTP4    DS    PL8    X'02C0'
EXTUSRE  EQU   *-EXTDATA
*
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
EXTRECNT DS    XL4    X'02C8'    PERRECNT      (131)  N N N Y Y Y Y Y
EXTROWID DS    CL8    X'02CC'    RMUOWID       (132)  N N N Y Y Y Y Y
EXTSCLS  DS    CL8    X'02D4'    SRVCLSNM      (167)  N N N Y Y Y Y Y
EXTRCLS  DS    CL8    X'02DC'    RPTCLSNM      (168)  N N N Y Y Y Y Y
EXTFNME  DS    CL4    X'02E4'    FCTYNAME      (163)  N N N Y Y Y Y Y
EXTTFLG  DS    XL8    X'02E8'    TRANFLAG      (164)  N N N Y Y Y Y Y
EXTTINF  DS    XL4    X'02F0'    TERMINFO      (165)  N N N Y Y Y Y Y
```

```
EXTCNNM  DS    CL4    X'02F4'    TERMCNNM      (169)  N N N Y Y Y Y Y
EXTSGCT  DS    XL4    X'02F8'    SC24SGCT      (144)  N N N Y Y Y Y Y
EXTGSHR  DS    XL4    X'02FC'    SC24GSHR      (145)  N N N Y Y Y Y Y
EXTFSHR  DS    XL4    X'0300'    SC24FSHR      (146)  N N N Y Y Y Y Y
EXTSGC31 DS    XL4    X'0304'    SC31SGCT      (147)  N N N Y Y Y Y Y
EXTGSH31 DS    XL4    X'0308'    SC31GSHR      (148)  N N N Y Y Y Y Y
EXTFSH31 DS    XL4    X'030C'    SC31FSHR      (149)  N N N Y Y Y Y Y
EXTLOGW  DS    XL4    X'0310'    LOGWRTCT      (172)  N N N Y Y Y Y Y
EXTSYTM  DS    XL4    X'0314'    SYNCTIME      (173)  N N N Y Y Y Y Y
EXTRLSW  DS    XL4    X'0318'    RLSWAIT       (174)  N N N Y Y Y Y Y
*
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
EXTRLSC  DS    XL4    X'031C'    RLSCPUT       (175)  N N N N Y Y Y Y
EXTBRDG  DS    CL4    X'0320'    BRDGTRAN      (124)  N N N N Y Y Y Y
EXTRMCT  DS    XL4    X'0324'    PCLURMCT      (072)  N N N N Y Y Y Y
EXTTOTCT DS    XL4    X'0328'    ICTOTCT       (066)  N N N N Y Y Y Y
EXTLMDE  DS    XL4    X'032C'    LMDELAY       (128)  N N N N Y Y Y Y
EXTWTEX  DS    XL4    X'0330'    WTEXWAIT      (181)  N N N N Y Y Y Y
EXTWTCE  DS    XL4    X'0334'    WTCEWAIT      (182)  N N N N Y Y Y Y
EXTICDE  DS    XL4    X'0338'    ICDELAY       (183)  N N N N Y Y Y Y
EXTGVUP  DS    XL4    X'033C'    GVUPWAIT      (184)  N N N N Y Y Y Y
EXTTSSW  DS    XL4    X'0340'    TSSHWAIT      (178)  N N N N Y Y Y Y
*
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
EXTURID  DS    CL16   X'0344'    RRMSURID      (190)  N N N N N Y Y Y
EXTPRCSN DS    CL36   X'0354'    PRCSNAME      (200)  N N N N N Y Y Y
EXTSTYPE DS    CL8    X'0378'    PRCSTYPE      (201)  N N N N N Y Y Y
EXTCSID  DS    CL52   X'0380'    PRCSID        (202)  N N N N N Y Y Y
EXTTYID  DS    CL52   X'03B4'    ACTVTYID      (203)  N N N N N Y Y Y
EXTTYNM  DS    CL16   X'03E8'    ACTVTYNM      (204)  N N N N N Y Y Y
EXTCLIP  DS    CL16   X'03F8'    CLIPADDR      (244)  N N N N N Y Y Y
EXTPLCT  DS    XL4    X'0408'    PCDPLCT       (073)  N N N N N Y Y Y
EXTPICT  DS    XL4    X'040C'    CFCAPICT      (025)  N N N N N Y Y Y
EXTRSYN  DS    XL4    X'0410'    BARSYNCT      (205)  N N N N N Y Y Y
EXTSYCT  DS    XL4    X'0414'    BARASYCT      (206)  N N N N N Y Y Y
EXTPACT  DS    XL4    X'0418'    BALKPACT      (207)  N N N N N Y Y Y
EXTPROC  DS    XL4    X'041C'    BADPROCT      (208)  N N N N N Y Y Y
EXTCTCT  DS    XL4    X'0420'    BADACTCT      (209)  N N N N N Y Y Y
EXTSPAC  DS    XL4    X'0424'    BARSPACT      (210)  N N N N N Y Y Y
EXTUPAC  DS    XL4    X'0428'    BASUPACT      (211)  N N N N N Y Y Y
EXTMPAC  DS    XL4    X'042C'    BARMPACT      (212)  N N N N N Y Y Y
EXTCPAC  DS    XL4    X'0430'    BADCPACT      (213)  N N N N N Y Y Y
EXTOPCT  DS    XL4    X'0434'    BAACQPCT      (214)  N N N N N Y Y Y
```

```
EXTTPCT  DS    XL4    X'0438'    BATOTPCT      (215)  N N N N N Y Y Y
EXTRDCC  DS    XL4    X'043C'    BAPRDCCT      (216)  N N N N N Y Y Y
EXTCDCC  DS    XL4    X'0440'    BAACDCCT      (217)  N N N N N Y Y Y
EXTOTCC  DS    XL4    X'0444'    BATOTCCT      (218)  N N N N N Y Y Y
EXTATEC  DS    XL4    X'0448'    BARATECT      (219)  N N N N N Y Y Y
EXTIECT  DS    XL4    X'044C'    BADFIECT      (220)  N N N N N Y Y Y
EXTAECT  DS    XL4    X'0450'    BATIAECT      (221)  N N N N N Y Y Y
EXTTECT  DS    XL4    X'0454'    BATOTECT      (222)  N N N N N Y Y Y
EXTCVCT  DS    XL4    X'0458'    WBRCVCT       (231)  N N N N N Y Y Y
EXTHRIN  DS    XL4    X'045C'    WBCHRIN       (232)  N N N N N Y Y Y
EXTNDCT  DS    XL4    X'0460'    WBSENDCT      (233)  N N N N N Y Y Y
EXTROUT  DS    XL4    X'0464'    WBCHROUT      (234)  N N N N N Y Y Y
EXTWBTO  DS    XL4    X'0468'    WBTOTCT       (235)  N N N N N Y Y Y
EXTPRCT  DS    XL4    X'046C'    WBREPRCT      (236)  N N N N N Y Y Y
EXTPWCT  DS    XL4    X'0470'    WBREPWCT      (237)  N N N N N Y Y Y
EXTRECT  DS    XL4    X'0474'    DHCRECT       (226)  N N N N N Y Y Y
EXTNSCT  DS    XL4    X'0478'    DHINSCT       (227)  N N N N N Y Y Y
EXTETCT  DS    XL4    X'047C'    DHSETCT       (228)  N N N N N Y Y Y
EXTRETC  DS    XL4    X'0480'    DHRETCT       (229)  N N N N N Y Y Y
EXTOTCT  DS    XL4    X'0484'    DHTOTCT       (230)  N N N N N Y Y Y
EXTTDCL  DS    XL4    X'0488'    DHTOTDCL      (240)  N N N N N Y Y Y
EXTENCT  DS    XL4    X'048C'    SOBYENCT      (242)  N N N N N Y Y Y
EXTYDEC  DS    XL4    X'0490'    SOBYDECT      (243)  N N N N N Y Y Y
EXTEQCT  DS    XL4    X'0494'    IMSREQCT      (179)  N N N N N Y Y Y
EXT2REQ  DS    XL4    X'0498'    DB2REQCT      (180)  N N N N N Y Y Y
EXTDECT  DS    XL4    X'049C'    CHMODECT      (248)  N N N N N Y Y Y
EXTTTCT  DS    XL4    X'04A0'    TCBATTCT      (251)  N N N N N Y Y Y
EXTISPT  DS    XL4    X'04A4'    QRDISPT       (255)  N N N N N Y Y Y
EXTRCPU  DS    XL4    X'04A8'    QRCPUT        (256)  N N N N N Y Y Y
EXTSDIS  DS    XL4    X'04AC'    MSDISPT       (257)  N N N N N Y Y Y
EXTSCPU  DS    XL4    X'04B0'    MSCPUT        (258)  N N N N N Y Y Y
EXTL8CP  DS    XL4    X'04B4'    L8CPUT        (259)  N N N N N Y Y Y
EXTJ8CP  DS    XL4    X'04B8'    J8CPUT        (260)  N N N N N Y Y Y
EXTS8CP  DS    XL4    X'04BC'    S8CPUT        (261)  N N N N N Y Y Y
EXTDDLY  DS    XL4    X'04C0'    QRM0DDLY      (249)  N N N N N Y Y Y
EXTTDLY  DS    XL4    X'04C4'    MAXOTDLY      (250)  N N N N N Y Y Y
EXTELAY  DS    XL4    X'04C8'    GNQDELAY      (123)  N N N N N Y Y Y
EXTFDTW  DS    XL4    X'04CC'    CFDTWAIT      (176)  N N N N N Y Y Y
EXTYWTT  DS    XL4    X'04D0'    SRVSYWTT      (177)  N N N N N Y Y Y
EXTRMSW  DS    XL4    X'04D4'    RRMSWAIT      (191)  N N N N N Y Y Y
EXTRMTT  DS    XL4    X'04D8'    RUNTRWTT      (195)  N N N N N Y Y Y
EXTCDLY  DS    XL4    X'04DC'    SYNCDLY       (196)  N N N N N Y Y Y
EXTOWTT  DS    XL4    X'04E0'    SOIOWTT       (241)  N N N N N Y Y Y
EXTMSWA  DS    XL4    X'04E4'    IMSWAIT       (186)  N N N N N Y Y Y
EXTDYQW  DS    XL4    X'04E8'    DB2RDYQW      (187)  N N N N N Y Y Y
EXTONWT  DS    XL4    X'04EC'    DB2CONWT      (188)  N N N N N Y Y Y
EXT2WAI  DS    XL4    X'04F0'    DB2WAIT       (189)  N N N N N Y Y Y
EXTMTIM  DS    XL4    X'04F4'    JVMTIME       (253)  N N N N N Y Y Y
EXTMSUS  DS    XL4    X'04F8'    JVMSUSP       (254)  N N N N N Y Y Y
```

```
EXTGRPID DS    CL28   X'04FC'    TRNGRPID      (082)  N N N N N Y Y Y
*
* TRANSACTION SERVER 2.1 FIELDS
*
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
EXTRQRWT DS    XL4    X'0518'    RQRWAIT       (192)  N N N N N N Y Y
EXTRQPWT DS    XL4    X'051C'    RQPWAIT       (193)  N N N N N N Y Y
EXTTSTID DS    CL128  X'0520'    OTSTID        (194)  N N N N N N Y Y
EXTNETID DS    CL8    X'05A0'    NETID         (197)  N N N N N N Y Y
EXTRLUNM DS    CL8    X'05A8'    RLUNAME       (198)  N N N N N N Y Y
EXTTSINW DS    XL4    X'05B0'    OTSINDWT      (199)  N N N N N N Y Y
EXTWBRD  DS    XL4    X'05B4'    WBREADCT      (224)  N N N N N N Y Y
EXTWBWR  DS    XL4    X'05B8'    WBWRITCT      (225)  N N N N N N Y Y
EXTWBXTR DS    XL4    X'05BC'    WBEXTRCT      (238)  N N N N N N Y Y
EXTWBRW  DS    XL4    X'05C0'    WBBRWCT       (239)  N N N N N N Y Y
EXTTCPSV DS    CL8    X'05C4'    TCPSRVCE      (245)  N N N N N N Y Y
EXTPORT  DS    XL4    X'05CC'    PORTNUM       (246)  N N N N N N Y Y
EXTKY8DT DS    XL4    X'05D0'    KY8DISPT      (262)  N N N N N N Y Y
EXTKY8CT DS    XL4    X'05D4'    KY8CPUT       (263)  N N N N N N Y Y
EXTJVMIT DS    XL4    X'05D8'    JVMITIME      (273)  N N N N N N Y Y
EXTJVMRT DS    XL4    X'05DC'    JVMRTIME      (275)  N N N N N N Y Y
EXTSOXTR DS    XL4    X'05E0'    SOEXTRCT      (289)  N N N N N N Y Y
EXTSONPS DS    XL4    X'05E4'    SOCNPSCT      (290)  N N N N N N Y Y
EXTSOPS  DS    XL4    X'05E8'    SOCPSCT       (291)  N N N N N N Y Y
EXTSONSH DS    XL4    X'05EC'    SONPSHWM      (292)  N N N N N N Y Y
EXTSOPSH DS    XL4    X'05F0'    SOPSHWM       (293)  N N N N N N Y Y
EXTSORCV DS    XL4    X'05F4'    SORCVCT       (294)  N N N N N N Y Y
EXTSOCIN DS    XL4    X'05F8'    SOCHRIN       (295)  N N N N N N Y Y
EXTSOSND DS    XL4    X'05FC'    SOSENDCT      (296)  N N N N N N Y Y
EXTSOCOT DS    XL4    X'0600'    SOCHROUT      (297)  N N N N N N Y Y
EXTSOTOT DS    XL4    X'0604'    SOTOTCT       (298)  N N N N N N Y Y
EXTSOIOW DS    XL4    X'0608'    SOOIOWTT      (299)  N N N N N N Y Y
*
*
* TRANSACTION SERVER 2.2 FIELDS
*
*                                                     2 3 4 5 5 5 6 6
*                                                     1 3 1 1 2 3 1 2
*                     EXTOFF     FLDNAME       FLDID  2 0 0 0 0 0 0 0
*                     ------     -----------   -----  - - - - - - - -
EXTMXJTD DS    XL4    X'060C'    MAXJTDLY      (277)  N N N N N N N Y
EXTMXHTD DS    XL4    X'0610'    MAXHTDLY      (278)  N N N N N N N Y
EXTPTPWT DS    XL4    X'0614'    PTPWAIT       (285)  N N N N N N N Y
EXTSOMI1 DS    XL4    X'0618'    SOMSGIN1      (301)  N N N N N N N Y
EXTSOCI1 DS    XL4    X'061C'    SOCHRIN1      (302)  N N N N NJSJSA002
EXTSOMO1 DS    XL4    X'0620'    SOMSGOU1      (303)  N N N N NSCJSA001
```

```
EXTSOCO1 DS    XL4    X'0624'    SOCHROU1      (304)  N N N N NJSJSA002
*                                                              SCJSA001
* TRANSACTION SERVER 2.3 FIELDS                                SCJSA001
* Plus 5 from 3.1                                              JSJSA002
*                                               5 5 5 6 6 6 6  JSJSA002
*                                               1 2 3 1 2 3 4  JSJSA002
*                    EXTOFF  FLDNAME     FLDID  0 0 0 0 0 0 0  JSJSA002
*                    ------  ---------   -----  - - - - - - -  JSJSA002
EDSTCBHW DS    XL4  X'0628'  DSTCBHWM    (252)  N N N N N Y    SCJSA001
EKY9DISP DS    XL4  X'062C'  KY9DISPT    (264)  N N N N N Y    JSJSA002
EPCLNKCC DS    XL4  X'0630'  PCLNKCCT    (306)  N N N N N N Y  JSJSA002
EKY9CPUT DS    XL4  X'0634'  KY9CPUT     (265)  N N N N N Y    JSJSA002
EPCXCLCC DS    XL4  X'0638'  PCXCLCCT    (307)  N N N N N N Y  JSJSA002
EJ9CPUT  DS    XL4  X'063C'  J9CPUT      (267)  N N N N N Y    JSJSA002
EPCDPLCC DS    XL4  X'0640'  PCDPLCCT    (308)  N N N N N N Y  JSJSA002
EDSTCBMW DS    XL4  X'0644'  DSTCBMWT    (268)  N N N N N Y    JSJSA002
EPCRTNCC DS    XL4  X'0648'  PCRTNCCT    (309)  N N N N N N Y  JSJSA002
EDSMMSCW DS    XL4  X'064C'  DSMMSCWT    (279)  N N N N N Y    JSJSA002
EPCRTNCD DS    XL4  X'0650'  PCRTNCDL    (310)  N N N N N N Y  JSJSA002
ECBSRVRN DS    CL4  X'0654'  CBSRVRNM    (311)  N N N N N Y    SCJSA001
EEJBSACC DS    XL4  X'0658'  EJBSACCT    (312)  N N N N N Y    SCJSA001
EEJBSPAC DS    XL4  X'065C'  EJBSPACT    (313)  N N N N N Y    SCJSA001
EEJBCREC DS    XL4  X'0660'  EJBCRECT    (314)  N N N N N Y    SCJSA001
EEJBREMC DS    XL4  X'0664'  EJBREMCT    (315)  N N N N N Y    SCJSA001
EEJBMTCH DS    XL4  X'0668'  EJBMTHCT    (316)  N N N N N Y    SCJSA001
EEJBTOTC DS    XL4  X'066C'  EJBTOTCT    (317)  N N N N N Y    SCJSA001
*                                                              JSJSA001
* TRANSACTION SERVER 3.1 FIELDS                                JSJSA001
*                                                              JSJSA001
*                                               5 5 6 6 6 6    JSJSA001
*                                               2 3 1 2 3 4    JSJSA001
*                    EXTOFF  FLDNAME     FLDID  0 0 0 0 0 0    JSJSA001
*                    ------  ---------   -----  - - - - - -    JSJSA001
EL9CPUT  DS    XL4  X'0670'  L9CPUT      (266)  N N N N N Y    JSJSA002
EMAXSTDL DS    XL4  X'0674'  MAXSTDLY    (281)  N N N N N Y    JSJSA002
EDSCHMDL DS    XL4  X'0678'  DCSHMDLY    (247)  N N N N N Y    JSJSA002
EICSTRCC DS    XL4  X'067C'  ICSTRCCT    (346)  N N N N N Y    JSJSA002
EICSTRCD DS    XL4  X'0680'  ICSTRCDL    (347)  N N N N N Y    JSJSA002
EWBREDOC DS    XL4  X'0684'  WBREDOCT    (331)  N N N N N Y    JSJSA002
EWBWRTOC DS    XL4  X'0688'  WBWRTOCT    (332)  N N N N N Y    JSJSA002
EWBRCVIN DS    XL4  X'068C'  WBRCVIN1    (333)  N N N N N Y    JSJSA002
EWBCHRIN DS    XL4  X'0690'  WBCHRIN1    (334)  N N N N N Y    JSJSA002
EWBSNDOU DS    XL4  X'0694'  WBSNDOU1    (335)  N N N N N Y    JSJSA002
EWBCHROU DS    XL4  X'0698'  WBCHROU1    (336)  N N N N N Y    JSJSA002
EWBPARSC DS    XL4  X'069C'  WBPARSCT    (337)  N N N N N Y    JSJSA002
EWBBRWOC DS    XL4  X'06A0'  WBBRWOCT    (338)  N N N N N Y    JSJSA002
EWBIWBSC DS    XL4  X'06A4'  WBIWBSCT    (340)  N N N N N Y    JSJSA002
EWBREPRD DS    XL4  X'06A8'  WBREPRDL    (341)  N N N N N Y    JSJSA002
EWBREPWD DS    XL4  X'06AC'  WBREPWDL    (342)  N N N N N Y    JSJSA002
```

```
EICSTACC DS    XL4  X'06B0'  ICSTACCT    (065)  N N N N N Y    JSJSA002
EPGTOTCC DS    XL4  X'06B4'  PGTOTCCT    (321)  N N N N N Y    JSJSA002
EPGBRWCC DS    XL4  X'06B8'  PGBRWCCT    (322)  N N N N N Y    JSJSA002
EPGGETCC DS    XL4  X'06BC'  PGGETCCT    (323)  N N N N N Y    JSJSA002
EPGPUTCC DS    XL4  X'06C0'  PGPUTCCT    (324)  N N N N N Y    JSJSA002
EPGMOVCC DS    XL4  X'06C4'  PGMOVCCT    (325)  N N N N N Y    JSJSA002
EPGGETCD DS    XL4  X'06C8'  PGGETCDL    (326)  N N N N N Y    JSJSA002
EPGPUTCD DS    XL4  X'06CC'  PGPUTCDL    (327)  N N N N N Y    JSJSA002
EPGCRECC DS    XL4  X'06D0'  PGCRECCT    (328)  N N N N N Y    JSJSA002
EPCDLCSD DS    XL4  X'06D4'  PCDLCSDL    (286)  N N N N N Y    JSJSA002
EICSTACD DS    XL4  X'06D8'  ICSTACDL    (345)  N N N N N Y    JSJSA002
EPCDLCRD DS    XL4  X'06DC'  PCDLCRDL    (287)  N N N N N Y    JSJSA002
ERODISPT DS    XL4  X'06E0'  RODISPT     (269)  N N N N N Y    JSJSA002
EROCPUT  DS    XL4  X'06E4'  ROCPUT      (270)  N N N N N Y    JSJSA002
EX8CPUT  DS    XL4  X'06E8'  X8CPUT      (271)  N N N N N Y    JSJSA002
EX9CPUT  DS    XL4  X'06EC'  X9CPUT      (272)  N N N N N Y    JSJSA002
EMAXXTDL DS    XL4  X'06F0'  MAXXTDLY    (282)  N N N N N Y    JSJSA002
*                                                              JSJSA002
*- DFHRMI Fields.  Note that FLDID is no longer unique.        DZJSA001
*                                               5 5 6 6 6 6    DZJSA001
*                                               2 3 1 2 3 4    DZJSA001
*                    EXTOFF  FLDNAME     FLDID  0 0 0 0 0 0    DZJSA001
*                    ------  ---------   -----  - - - - - -    DZJSA001
EMITOTAL DS    XL4  X'06F4'  RMITOTAL    (001)  N N N N N Y    DZJSA001
EMIOTHER DS    XL4  X'06F8'  RMIOTHER    (002)  N N N N N Y    DZJSA001
EMIDB2DL DS    XL4  X'06FC'  RMIDB2LY    (003)  N N N N N Y    DZJSA001
EMIDBCTL DS    XL4  X'0700'  RMIDBCTL    (004)  N N N N N Y    DZJSA001
EMIEXDLI DS    XL4  X'0704'  RMIEXDLI    (005)  N N N N N Y    DZJSA001
EMIMQMDL DS    XL4  X'0708'  RMIMQMLY    (006)  N N N N N Y    DZJSA001
EMICPSML DS    XL4  X'070C'  RMICPSMY    (007)  N N N N N Y    DZJSA001
EMITCPIP DS    XL4  X'0710'  RMITCPIP    (008)  N N N N N Y    DZJSA001
*                                                              DZJSA003
*- CTS 3.2 Drop 4 Fields                                       DZJSA003
*                                               5 6 6 6 6 6    DZJSA003
*                                               3 1 2 3 4 5    DZJSA003
*                    EXTOFF  FLDNAME     FLDID  0 0 0 0 0 0    DZJSA003
*                    ------  ---------   -----  - - - - - -    DZJSA003
EDHDELCT DS    XL4  X'714'   DHDELCT     (223)  N N N N N Y    DZJSA003
EISALLOC DS    XL4  X'718'   ISALLOCT    (288)  N N N N N Y    DZJSA003
ECLIPPOR DS    XL4  X'71C'   CLIPPORT    (330)  N N N N N Y    DZJSA003
EOPORTNU DS    XL4  X'720'   OPORTNUM    (367)  N N N N N Y    DZJSA003
EOCLIPOR DS    XL4  X'724'   OCLIPORT    (369)  N N N N N Y    DZJSA003
EOTRANFL DS    CL8  X'728'   OTRANFLG    (370)  N N N N N Y    DZJSA003
EISIPCNN DS    CL8  X'730'   ISIPCNNM    (305)  N N N N N Y    DZJSA003
EOAPPLID DS    CL8  X'738'   OAPPLID     (360)  N N N N N Y    DZJSA003
EOTRAN   DS    CL4  X'740'   OTRAN       (363)  N N N N N Y    DZJSA003
EOUSERID DS    CL8  X'744'   OUSERID     (364)  N N N N N Y    DZJSA003
EOUSERCO DS    CL64 X'74C'   OUSERCO     (365)  N N N N N Y    DZJSA003
EOTCPSVC DS    CL8  X'78C'   OTCPSVC     (366)  N N N N N Y    DZJSA003
```

```
EOCLIPAD DS    CL16 X'794'   OCLIPAD     (368)  N N N N N Y    DZJSA003
EOFCTYNM DS    CL8  X'7A4'   OFCTYNM     (371)  N N N N N Y    DZJSA003
EOTRANNU DS    PL4  X'7AC'   OTRANNUM    (362)  N N N N N Y    DZJSA003
EISIOWTT DS    XL4  X'7B0'   ISIOWTT     (300)  N N N N N Y    DZJSA003
EOSTART  DS    CL8  X'7B4'   OSTART      (361)  N N N N N Y    DZJSA003
*--      CTS 3.2 Drop 5 additions                              DZJSA003
EPGCSTHW DS    XL4  X'7BC'   PGCSTHWM    (329)  N N N N N Y    DZJSA003
EWMQREQC DS    XL4  X'7C0'   WMQREQCT    (395)  N N N N N Y    DZJSA003
EONETWKI DS    CL8  X'7C4'   ONETWKID    (359)  N N N N N Y    DZJSA003
EWMQGETW DS    xL4  X'7CC'   WMQGETWT    (396)  N N N N N Y    DZJSA003
EXTBDY   EQU   *-EXTTRAN
EXTLEN   EQU   *-EXTDATA
***********************************************************************
```

CICS CMF Dynamic Dictionary Support

Dynamic dictionary support removes a limitation of static dictionary support. The static dictionary had to be changed manually if you made a change to the MCT that affected the CICS dictionary.

Dynamic dictionary support remaps the CICS dictionary to the CA JARS dictionary whenever a new CICS dictionary is encountered in the SMF input stream. This eliminates any incompatibility between the CICS dictionary and the CA JARS dictionary.

Enabling the Dynamic Dictionary

There are no new CA JARS control statements required.

Dynamic dictionary support is enabled by adding three mandatory and one optional DD cards to the CA JARS JCL.

The mandatory DD cards are //CAIJSDIN for CAJ1CMU3 control statements, //SYSPRINT for CAJ1CMU3 messages, and //CAJRCTMP, which is used to pass the CA JARS dictionary updates back to CA JARS main processing from CAJ1CMU3. The recommended form of these statements is identified by arrows in the JCL below.

The optional DD statement, //CAIJSDCO, is discussed in the section, Saving Dynamic Dictionaries.

//CAIJSDIN in the JCL activates dynamic dictionary support:

- If //CAIJSDIN is coded, //SYSPRINT and //CAJRCTMP must be coded.
- If //CAIJSDIN is not coded, //SYSPRINT and //CAJRCTMP are ignored.

```
//JARS  EXEC PGM=JSIMAIN,REGION=4M
  .
  .
  .
 //*  BUILD THE JARS DICTIONARY FROM THE CICS DICTIONARY
//*
//CAIJSDCI DD DISP=SHR,DSN=CAI.JARS.DYNAMDIC(0)
 //SYSPRINT DD SYSOUT=*                                   <===
//CAJRCTMP DD DISP=(,PASS),UNIT=SYSDA,SPACE=(TRK,(5,5)),  <===
//            DCB=BLKSIZE=4080                            <===
//CAIJSDIN DD *                                           <===
    SYSID=ALL
    APPLID=ALL
    USERFLD=RMIDATA,C6L16
/*
```

Even with dynamic dictionary support enabled, CA JARS must be started with a populated CA JARS dictionary in the dictionary file //CAIJSDCI. This is because there is no guarantee the first SMF 110 record encountered in a run will be the appropriate CICS dictionary SMF record. Monitor records that do not match a dictionary entry are discarded by CA JARS.

Reviewing Processing Options

At the end of the CA JARS run, the original dictionary file or files are left intact. You can either leave them as is, replace them with the dynamic dictionary file, or update portions of the CA JARS dictionary. CA JARS will write in-storage dictionaries to the optional DD statement //CAJSDCO. CA JARS will select the **last** instance of each unique SYSID/APPLID/CICS release key as the current dictionary.

The user's options for managing CA JARS dictionaries depend on the complexity of the CICS configurations. If all instances of CICS use the default MCT or at least use an identical MCT, then a single CA JARS dictionary will do for all. (This is equivalent to coding SYSID=ALL and APPLID=ALL as the CAJ1CMU3 control statements.) In this case, this sample CAJ1CMU3 control statements may do:

```
//CAIJSDIN  DD *         CAJ1CMU3 Control Statements
    SYSID=ALL
    APPLID=ALL

/*
```

If different systems use different MCTs, do not use wildcards with SYSID and APPLID.

Note: The CICS release cannot use wildcards because releases of CICS do not have compatible dictionaries.

The more complex case, in which you maintain multiple, incompatible MCTs, requires CA JARS to maintain multiple dictionaries. Dictionaries are identified by SYSID and/or APPLID. As always, CA JARS reads //CAIJSDCI at the beginning to "prime" the in-storage dictionary table. Every SMF 110 CICS dictionary record encountered in the input stream is reformatted as a CA JARS dictionary and written to //CAJRCTMP. CA JARS then rereads CAJRCTMP and replaces each in-storage dictionary with the last instance of itself encountered in CAJRCTMP. Instances are identified by SYSID, APPLID, and CICS release.

The file //CAIJSDCI must contain at least one CA JARS dictionary, and it may contain multiple dictionaries. It may point to a concatenation of data sets containing dictionaries. See DSN=CAI.JARS.SAMPLIB(DYNDUCGD) for a suggested utility to define the CA JARS dynamic dictionary GDG, and "prime" the first generation.

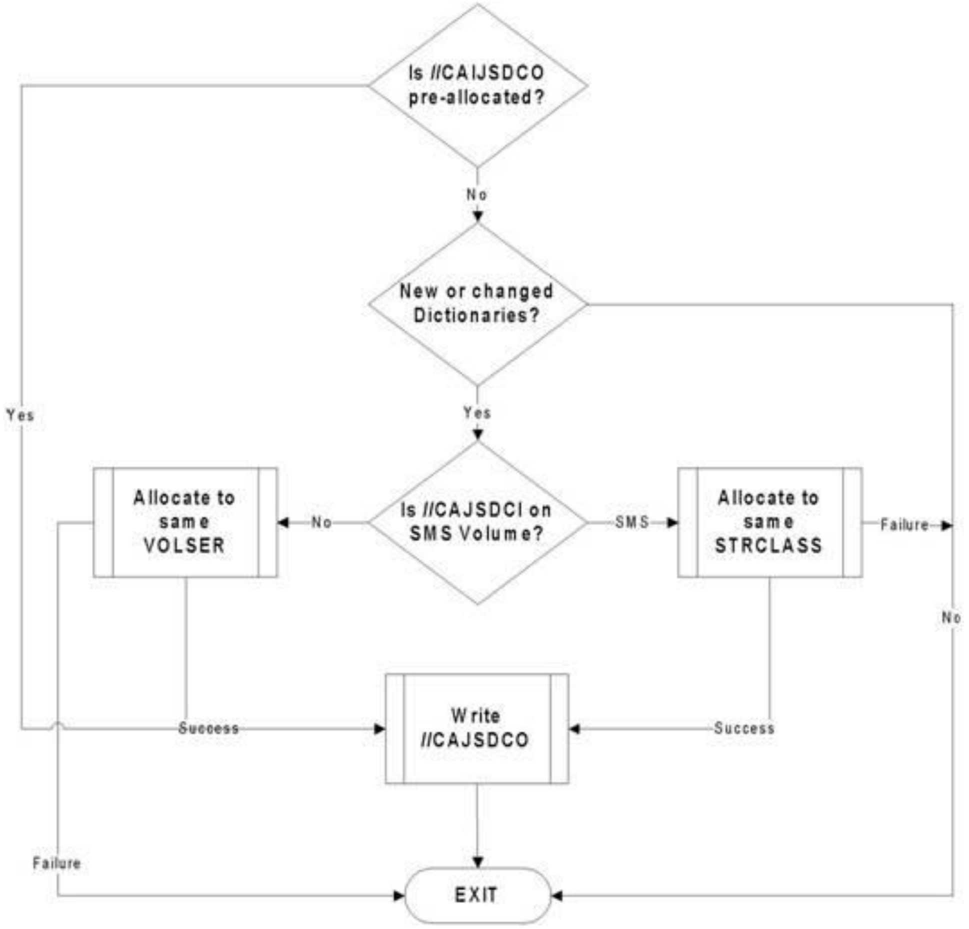
If you want, CA JARS can write all current dictionaries at the end of the run to a single OS data set through the file //CAIJSDCO. See the next section for processing details.

Saving Dynamic Dictionaries

If the optional DD statement //CAIJSDCO is coded in the CA JARS JCL, CA JARS will always write the dictionary or dictionaries out at the end of run, in their final state.

CA recommends the input dictionary //CAIJSDCI be coded as a GDG. Then CA JARS will write only new or changed dictionaries to a new GDG generation.

CA JARS will use dynamic allocation to create a new GDG generation to which to write the changed dictionary, as illustrated in this flowchart:


Is //CAIJSDCO pre-allocated?
No
New or changed Dictionaries?
Yes
Yes
Is //CAJSDCI on SMS Volume?
No
SMS
Allocate to same VOLSER
Allocate to same STRCLASS
Failure
No
Success
Write //CAJSDCO
Success
Failure
EXIT

A new generation will only be created if the in-storage dictionary has changed, or a new dictionary has been added. CA recommends the GDG-input, dynamically allocated output option as it is somewhat more efficient to only write output when changed.

Note: If the input CA JARS dictionary is on an SMS-controlled volume, CA JARS will allocate the output dictionary data set to the same STRCLASS. IF //CAIJSDCI is on a non-SMS volume, CA JARS will attempt to allocate //CAIJSDCO to the same volume: UNIT=3390,VOL=SER=(volser of CAIJSDCI).

Appendix A: CA JARS Exits

The function of this type 2 exit is to produce reports describing changes to be made to RATE, TSORATE, FORMRATE, and RJERATE statements in order to achieve cost recovery goals for the following: TSO, CICS, VM, and Normal/Batch Rate Billing.

For the purpose of this exit, the basic I/O rate field on the RATE and/or TSORATE statement is ignored. The I/O factor fields have been redefined to mean I/O charge per 1000 EXCPs. All other fields on the RATE statements retain their original definitions. To implement the adjusted rate values computed by this exit, it is necessary to use the optional method of calculating I/O rate as described in your User Guide.

This exit requires that an additional control statement input file, ddname SJEX201I, and a report output file, ddname SJEX201O, be added to the JCL statements; each one defining a CPU ID, a type of recovery goal, and the recovery goal amount. The output file contains a diagnostic listing of all input statistics and the rate determination reports. An overview of the processing is as follows:

- on the first invocation of SJEX201, work areas are acquired
- all statements from SJEX201I are processed, and
- the diagnostic listing is printed.

Data is collected for a particular CPU-ID/Process-ID combination if it is to be used in the rate determination analysis. When all account records have been processed, the rate determination analysis reports are furnished and the work tables are released. If problems are encountered in the initialization phase, such as no recovery statements, then the data collection process is bypassed for all account records and rate determination reports will not be printed.

Note: If the rate determination exit is processed with a blank CPU-ID in the rate and recover cards, the only report which is produced is a list of the recovery control cards. A CPU-ID must be indicated to produce the rate determination analysis. For every different CPU-ID with the supplied recovery goals, there should be a corresponding appropriate RATE and/or TSORATE statement present. There must be matching FORMRATE and RJERATE statements for each combination of CPU-ID, SYSOUT CLASS, FORM-ID or CPU-ID, and RJE-LINE Name specified on recovery goals statements or no rate determination report is printed for that combination. PRIORITY control statements are ignored and no attempt is made to allow for the use of priority or job class factors.

It should also be noted that the rate determination reports are valid only for the data presented for that run. At best, this exit can only be used as a guide in determining future rate parameters by noting adjustments needed for the given data to produce given cost goals. This exit can in no way change the rate parameters used by CA JARS to compute an actual charge. As in the past, changes must be made to the RATE statements.

This section contains the following topics:

Operation

In order to use this type 2 exit for a rate determination analysis, it is necessary to place the name of this exit, SJEX201, on an EXITS control statement. In addition, RATE, TSORATE, FORMRATE, and/or RJERATE control statements must be supplied depending on the type of account/history file being analyzed. Two additional DD statements are necessary to define the exit's input control file (SJEX201I) and output report file (SJEX201O).

Example:

```
//SJEX201I  DD   *
        .
        .
        .
 RECOVERY CONTROL CARDS
        .
        .
        .
 //SJEX2010  DD   SYSOUT=A
//* DCB INFORMATION DEFINED IS LRECL=133 RECFM=FBA
//*                            BLKSIZE=133
```

Recovery goal RECOVER statements must be supplied. Each statement must contain:

- a CPU ID
- a recovery goal keyword defining the corresponding RATE statement entry
- the recovery goal amount

Optionally, a recovery goal CONTROL statement can be supplied limiting the number of CPU IDs analyzed or excluding a category. For example, no VM or TSO data. You accomplish this by setting those max CPU IDs to blanks or zeros.

If the rate determination exit is processed with a blank CPU ID in the rate and recover cards, the only report which is produced by the rate determination exit is a list of the recovery control cards. A CPU ID must be indicated to produce the rate determination analysis.

Rate Determination

For purposes of discussion of the rate determination process, items have been put into general groups. Each in turn can then be described as to the method of data collection and the procedure for determining an adjusted rate.

TIME:

PROCESSOR TIME

CONNECT TIME

A total time value is accumulated from the account/level 7 history records. Dividing the recovery goal by this value results in its ideal rate.

RECOVERY GOAL / TIME = IDEAL RATE

CORE:

CORE FACTOR

When an account/level 7 history record containing a core use value is found, then that core value is multiplied by the processing time for which that amount of core was used. This total core time value can then be divided into a recovery goal to set an ideal core factor rate.

RECOVERY GOAL / (CORE * TIME) = IDEAL RATE

MOUNTS:

TAPES USED

PRIVATE DISKS USED

Number of tape mounts and number of disk mounts are accumulated from the account/level 7 history records. Dividing the recovery goals by these totals result in ideal mount rates.

RECOVERY GOAL / TOTAL USED = IDEAL RATE

MAXIMUM STEP RATE

CA JARS can calculate step charges in two ways: the normal way, using Processor Time and Processor Rate or, optionally, using the following method:

CHARGE = MAXIMUM STEP RATE * STEP'S ELAPSED TIME

CA JARS always uses the method that results in the lower charge. Therefore, the maximum step rate is only invoked when its charge is less. When a maximum charge step record is found, the discount is determined by subtracting the actual charge from the total of all charges. The difference is accumulated and finally displayed with the count for the number of times applied, as well as the total number of step records processed as shown below:

TOTAL DISCOUNT = SUM (NORMAL CHARGE - STEP RATE CHARGE)

MINIMUM CHARGES:

- JOB
- SESSION
- TRANSACTION

When a minimum charge is applied, the difference between this value and the actual amount is considered to be a premium. This difference is accumulated and finally displayed with the number of times applied and with the total jobs/sessions/transactions processed.

I/O COUNTS:

- TEMPORARY STORAGE OPERATIONS
  - TAPE I/O COUNTS
- FILE OPERATIONS
  - DISK I/O COUNTS
- DL/1 OPERATIONS
  - OTHER I/O COUNTS
- TRANSIENT DATA OPERATIONS
  - TGET I/O COUNTS
  - TPUT I/O COUNTS

All of the above I/O counts are billed as a charge per 1000. Each data item is tallied and finally divided into the corresponding recovery goal to determine the ideal rate.

RECOVERY GOAL / (I/O COUNT / 1000) = IDEAL RATE

RJE SESSION CHARGES:

- CONNECT RATE
- EXCP RATE

Connect time durations and line transmissions (EXCPs) are accumulated for each combination of CPU-ID and LINE-NAME from RJE session account records. Ideal rates are calculated as follows:

RECOVERY GOAL / HOURS-OF-CONNECT-TIME = IDEAL RATE

RECOVERY GOAL / (EXCPs / 1000) = IDEAL RATE

RJE SURCHARGES:

TIME RATE

RECORD RATE

Reader and printer duration, Statements-Read and Lines-Printed / statements Punched Counts are accumulated for each CPU-ID and Input/Output Device-Name combination from each batch job. Since Line-Names and Input/Output destination names won't match, the RECOVERY statements specifying the appropriate recovery goals for these charges should either not specify a Line-Name or should specify each of the possible Input/Output device names in place of Line-Name. Ideal rates are calculated as follows:

RECOVERY GOALS / HOURS-OF-DURATION = IDEAL RATE

RECOVERY GOALS / ((CARDS-READ + LINES-PRINTED + CARDS PUNCHED)/ 1000) = IDEAL RATE

FORMS CHARGES:

LINES

PAGES

SETUP

Lines and Pages Printed and occurrences per job accumulated for each CPU-ID, SYSOUT-CLASS and FORMS-ID combination for local and remote destinations (remote destination implied by a nonzero output route code). Ideal rates for each combination are computed as follows:

Suppress charges for remote destinations not specified on matching FORMRATE statement:

RECOVERY GOAL / ((LOCAL-LINES + REMOTE-LINES) / 1000) = IDEAL RATE RECOVERY GOAL / (LOCAL-PAGES + REMOTE-PAGES) = IDEAL RATE RECOVERY GOAL / (LOCAL-SETUPS + REMOTE-SETUPS) = IDEAL RATE

Suppress charges for remote destinations specified on matching FORMRATE statement:

RECOVERY GOAL / (LOCAL-LINES / 1000) = IDEAL RATE

RECOVERY GOAL / LOCAL-PAGES = IDEAL RATE

RECOVERY GOAL / LOCAL-SETUPS = IDEAL RATE

Recovery Goal Keyword Tables

Normal Rate Recovery Goal Keywords

Keyword	Meaning
CPU	Basic CPU Processor
CORE	Core Factor
RDR	Non-spooled Reader I/O
PRT	Non-spooled Printer I/O
PUN	Non-spooled Punch I/O
TAP	Tape I/O
DSK	Disk I/O
OTH	Other I/O
TMOUNT	Tape Mount
DMOUNT	Disk Mount

TSO Rate Recovery Goal Keywords

Keyword	Meaning
TCPU	Basic CPU Processor
TCORE	Core Factor
TCONNECT	Connect Time
TRDR	Non-spooled Reader I/O
TPRT	Non-spooled Printer I/O
TPUN	Non-spooled Punch I/O
TTAP	Tape I/O
TDSK	Disk I/O
TOTH	Other I/O
TTPUTS	TPUTS I/O
TTGETS	TGETS I/O
TTMOUNT	Tape Mount
TDMOUNT	Disk Mount

CICS Rate Recovery Goal Keywords

Keyword	Meaning
CCPU	Basic CPU Processor
CCORE	Core Factor
CTSOPER	Temp Storage Operations
CFILEOPER	File Operations
CDL1OPER	DL/1 Operations
CTDOPER	Transient Date Operations

VM Rate Recovery Goal Keywords

Keyword	Meaning
VCPU	Basic CPU Processor
VCONNECT	Connect Time
VREADER	Spooled Reader I/O
VPRINTER	Spooled Print I/O
VPUNCH	Spooled Punch I/O
VNONSPOOL	Non-spooled I/O

FORM Rate Recovery Goal Keywords

Keyword	Meaning
FLINES	Line Rate
FPAGES	Page Rate
FSETUP	Setup Charge

RJE Rate Recovery Goal Keywords

Keyword	Meaning
RJTIME	Connect Rate
RJCOUNT	Transmission Rate
RSTIME	Time Rate

Keyword	Meaning
RSCOUNT	Record Rate

Error Numbers and Meaning

Number	Meaning
0	No Recover Statements Found
1	Total CPU IDs Value on Control statement is Zero
2	Invalid statement-type - Not Control or Recover
3	Invalid Recovery Goal Keyword
4	CPU ID Table is Full for Normal Rates
5	CPU ID Table is Full for TSO Rates
6	CPU ID Table is Full for CICS Rates
7	CPU ID Table is Full for VM Rates
8	No Matching RATE statements for Recover CPU IDs*
9	No Matching TSORATE statements for Recover CPU IDs*
10	No Matching CICS RATE statements for Recover CPU IDs*
11	No Matching VM TSORATE statements for Recover CPU IDs*
12	Account Data Matching Recovery Data not Found
13	CPU-ID/CLASS/FORMS-ID Table is Full for Forms Rates
14	CPU-ID/LINE-NAME Table is Full for RJE Rates
15	No Matching CPU-ID/CLASS/FORMS-ID FORMRATE statements for Recovery statements
16	No Matching CPU-ID/LINE-NAME RJERATE statements for Recovery statements
99	Program has been Modified with an Error Code 99

*Warning: A corresponding RATE or TSORATE statement was not found. The first RATE or TSORATE statement will be used instead.

Error Numbers for Forms Account Data Counters Overflow

Counter		Size
15	Local Lines	9(15)V
16	Remote Lines	9(15)V
17	Local Pages	9(15)V
18	Remote Pages	9(15)V
19	Local Setup	9(9)V
20	Remote Setup	9(9)V

Error Numbers for RJE Account Data Counters Overflow

Counter		Size
21	Reader/Printer Durations	9(10)V9(5)
22	Cards Read/Lines Printed	9(15)V
23	Session Connect Time	9(10)V9(5)
24	EXCP Count	9(15)V
25	No. of Minimum Charges	9(9)V
26	Difference (Calculated Charge, Minimum Charge)	9(13)V99

Error Numbers for Normal Account Data Counters Overflow

Counter		Size
30	Core Time	9(14)V9(5)
31	Number of Steps with a Maximum Charge	9(9)V
32	Total Lost Due to the Maximum Step Charge	9(13)V99
33	Processor	9(10)V9(5)
34	Cards Read	9(15)V
35	Lines Printed	9(15)V
36	Special Lines Printed	9(15)V
37	Cards Punched	9(15)V

Counter		Size
38	Reader EXCP I/O Count	9(15)V
39	Printer EXCP I/O Count	9(15)V
40	Punch I/O EXCP Count	9(15)V
41	TAPE I/O EXCP Count	9(15)V
42	Disk I/O EXCP Count	9(15)V
43	Other I/O EXCP Count	9(15)V
44	Tape Mounts	9(9)V
45	Disk Mounts	9(9)V
46	Number of Minimum Charge Jobs	9(9)V
47	Total Gained from Minimum Job Charges	9(13)V99
48	Total Number of Steps	9(15)V
49	Total Number of Jobs	9(15)V

Error Numbers for TSO Account Data Counters Overflow

Counter		Size
50	Core Time	9(14)V9(5)
51	Processor TIME	9(10)V9(5)
52	Connect Time	9(10)V9(5)
53	Spooled Lines Printed	9(15)V
54	Spooled Special Lines Printed	9(15)V
55	Spooled Cards Punched	9(15)V
56	Reader EXCP I/O Count	9(15)V
57	Printer EXCP I/O Count	9(15)V
58	Punch EXCP I/O Count	9(15)V
59	Tape EXCP I/O Count	9(15)V
60	Disk I/O EXCP Count	9(15)V
61	Other I/O EXCP Count	9(15)V
62	TPUT I/O EXCP Count	9(15)V
63	TGET I/O EXCP Count	9(15)v

Counter		Size
64	Tape Mounts	9(9)V
65	Disk Mounts	9(9)V
66	Number of Minimum Charge Jobs	9(9)V
67	Total Gained from Minimum Session Charges	9(13)V99
68	Total Number of Sessions	9(15)V

Error Numbers for CICS Account Data Counters Overflow

Counter		Size
70	Core Time	9(14)V9(5)
71	Processor Time	9(10)V9(5)
72	Temporary Storage Operations	9(15)V
73	File Operations	9(15)V
74	DL/1 Operations	9(15)V
75	Transient Data Operations	9(15)V
76	Number Of Minimum Charge Transactions	9(9)V
77	Total Gained From Minimum Transaction Charges	9(13)V
78	Total Transactions	9(15)V

Error Numbers for VM Account Data Counters Overflow

Counter		Size
80	Processor Time	9(10)V9(5)
81	Connect Time	9(10)V9(5)
82	Spooled Cards Read	9(15)V
83	Spooled Lines Printed	9(15)V
84	Spooled Cards Punched	9(15)V
85	Non-Spooled I/O Count	9(15)V
86	Total Number Of Sessions	9(9)V
87	Number Of Minimum Charge Sessions	9(15)V

Counter		Size
88	Total Gained From Minimum Sessions Charge	
9(13)V		

COMMENT Statement (SJEX201 Control Statement)

This statement is shown on the diagnostic report. All data on this statement, other than the asterisk in position 1, is ignored by this exit.

Position	Field Length	Field Name	Notes
1	1	Asterisk	Code an *
2-80	79	Comment	User note

CONTROL Statement (SJEX201 Control Statement)

The CONTROL statement is used to give the user some degree of control over the environment of this exit. If used, the CONTROL statement must be the first noncomment statement in the input file, and limits on the number of CPU IDs in each of the six categories may be supplied here. All nonnumeric data in the maximum CPU ID fields is changed to zeros before use by the exit. If a normal CA JARS report, or further processing of the account input file, is desired, the disposition field must contain the four letters PASS.

Position	Field Length	Field Name	Notes
1	1	Reserved	Not used
2-8	7	Function	'CONTROL'
9	1	Reserved	Not used
10-11	2	Maximum Number of CPU IDs for TSO Rate Analysis	99; right-justified
12	1	Reserved	Not used
13-14	2	Maximum Number of CPU IDs for TSO Rate Analysis	99; right-justified

Position	Field Length	Field Name	Notes
15	1	Reserved	Not used
16-17	2	Maximum Number of CPU IDs for TSO Rate Analysis	99; right-justified
18	1	Reserved	Not used
19-20	2	Maximum Number of CPU IDs for TSO Rate Analysis	99; right-justified
21	1	Reserved	Not used
22-23	2	Maximum Number of CPU-ID/FORMS-ID/ SYSOUT-CLASS Combinations for FORMRATE Analysis	99; right-justified
24	1	Reserved	Not used
25-26	2	Maximum Number of CPU-ID/LINE-NAME Combinations for RJERATE Analysis	99; right-justified
27-29	3	Reserved	Not used
30-33	4	Disposition	PASS to pass all Account Records to CA JARS for other reporting; otherwise all records are DELETED and the report is not produced.
34-80	4	Reserved	Not used

RECOVER Statement (SJEX201 Control Statement)

The RECOVER statement is used to supply a recovery goal amount for a particular category such as Basic Processor Charge for a particular CPU ID. All nonnumeric data in the recovery amount is converted to zeros before use by this exit.

Position	Field Length	Field Name	Notes
1	1	Reserved	Not used
2-8	7	Function	'RECOVER'
9	1	Reserved	Not used
10	1	CPU Identification	
11-16	6	Reserved	Not used
17-24	8	Recovery Goal Keyword	See Recovery Keyword Table
25	1	Reserved	Not used
26-34	9	Recovery Amount	999999999V; Note No Fractional Units. Right-justified
35	1	Reserved	Not used
36	1	SYSOUT CLASS	Matching FORMRATE statement
37	1	Reserved	Not used
38-41	4	FORMS-ID	Matching FORMRATE statement
42	1	Reserved	Not used
43-47	8	Line Name	Matching RJERATE statement
51-80	29	Reserved	Not used

Sample JCL To Remove Rate Determination Feature(s)

```
//RELINK   JOB  (ACCount),NAME,CLASS=A
//*
//*             REPLACE WITH A VALID JOB CARD FOR YOUR SHOP
//*
//LINKEDIT EXEC PGM=IEWL,PARM=(XREF,LET,LIST,NCAL),REGION=256K
//*
//SYSLMOD  DD   DSN=YOUR.LOADLIB,DISP=SHR
//SYSLIB   DD   DSN=YOUR.LOADLIB,DISP=SHR
//*                 ************
//*                 REPLACE WITH THE APPROPRIATE LIBRARY LOAD
                    MODULE DATA SET NAME
//*
//SYSUT1   DD   UNIT=SYSDA,SPACE=(1024,(50,20))
 //SYSPRINT DD   SYSOUT=A
//*
//SYSLIN   DD   *
     REPLACE  SJEX201V       **
     REPLACE  SJEX201X        ** REMOVE VM RATE PROCESSING MODULES
     REPLACE  SJEX201J       **
     INCLUDE  SYSLIB(SJEX201)
     NAME     SJEX201(R)
 //*
//*
//*        SJEX201N
//*        SJEX201R - TO REMOVE PROCESSING Of OS/DOS BATCH RATE INFO
//*        SJEX201K
//*
//*        SJEX201Z
//*        SJEX201X - TO REMOVE PROCESSING Of TSO RATE INFO
//*        SJEX201L
//*
//*        SJEX201I
//*        SJEX201W - TO REMOVE PROCESSING Of CICS RATE INFO
//*        SJEX201M
//*
//*        SJEX201V
//*        SJEX201X - TO REMOVE PROCESSING Of VM RATE INFO
//*        SJEX201J
//*
//*        SJEX2011 - TO REMOVE PROCESSING Of FORM RATE INFO
//*        SJEX2013
//*
//*        SJEX2014 - TO REMOVE PROCESSING Of RJE RATE INFO
//*        SJEX2016
//
```

SJEX201 Program Module Roster Summary Table

Program Module	Description
SJEX201	Driver for rate determination analysis
SJEX201C	Acquire and initialize the work tables
SJEX201D	Entry point in SJEX201C - free the work tables
SJEX201E	Add recovery data to the work tables
SJEX201F	Entry point in SJEX201G - open SJEX201I
SJEX201G	Read from SJEX201I, input control statement file
SJEX201H	Entry point in SJEX201G - close SJEX201I
SJEX201I	Collect CICS account data
SJEX201J	Print an expanded description of a VM Rate Table entry
SJEX201K	Print an expanded description of a normal Rate Table entry
SJEX201L	Print an expanded description of a TSO Rate Table entry
SJEX201M	Print an expanded description of a CICS Rate Table entry
SJEX201N	Collect normal account data
SJEX201O	Entry point in SJEX201P - open SJEX201O
SJEX201P	Write to SJEX201O, output listing file
SJEX201Q	Entry point in SJEX201P - Close SJEX201O
SJEX201R	Produce normal Rate Determination Analysis Reports
SJEX201S	Produce TSO Rate Determination Analysis Reports
SJEX201T	Duplicate the normal CA JARS report heading
SJEX201V	Collect VM account data
SJEX201W	Produce CICS Rate Determination Analysis Reports
SJEX201X	Produce VM Rate Determination Analysis Reports
SJEX201Y	Convert the System Julian Date (yyddd) to mm/dd/yy
SJEX201Z	Collect TSO account data
SJEX2011	Collect Forms account data

SPLITJOB Exit

The SPLITJOB exit lets you distribute the cost of a job among two or more cost centers. Split job processing requires the execution of two CA JARS reports. Report 1 (set code cannot be equal to X), computes the charges based on the rate statements given: FORMRATE, RATE, and TSORATE. It generates one CREDIT statement for the original job name and multiple DEBIT statements depending on the SPLIT TABLE control statement defined. This processing also requires a EXIT statement showing the SPLITJOB exit in position 18, as well as two additional DD statements:

- CONCOPY
- SPLITTBL

The exit suppresses all input to Report 1, causing it to be terminated. Report 2 which requires no exit routine, is then produced.

CA JARS Report 1

Purpose:

Supports the generation of DEBIT and CREDIT statements. Note that this report abends with error messages CAJS501I and CAJS9611.

- Additional, required CA JARS control statements:
  - Rate statements: RATE, TSORATE, FORMRATE
  - EXIT statement indicating SPLITJOB as EXIT2, position 18 **(SPLITJOB exit must be linked prior to execution)**
- Additional, required DD statements:

```
//CONCOPY DD DCB=(LRECL=80,BLKSIZE=800,RECFM=FB),UNIT=SYSDA,
                 VOL=SER=XXXXXX,DISP=(NEW,KEEP),DSN=XXXXX,
                 SPACE=(relative to the amount of input data)
//SPLITTBL DD *
```

- Split Table control statements to *follow* SPLITTBL DD statements:

Pos. 1

J or C
J if split based on job name
C if split based on account name

Pos. 2-9

eight positions; if job name, or
six positions; if account code with two blanks

Pos. 10-16

seven positions; the account to be charged

Pos. 17-19

three positions; percentage of split

CA JARS Report 2

Purpose:

Produces a true output report using the DEBIT and CREDIT statements generated by Report 1.

Report 2 uses the same rate statements specified for Report 1.

The SPLITJOB source code as distributed is set up to process account input. If you want to process history input, modifications to the SPLITJOB source code must be made. Refer to member SPLITJOB in CAJRSAMP for details.

Appendix B: The JR70CBF and JR70CB2 Glossaries for Combined Billing

The JR70CBF and JR70CB2 glossaries are provided primarily to process data from both the CA JARS z/OS and CA JARS VSE products for combined reporting via the Wizard Report Writer. However, their use is not limited to combined z/OS and VSE reporting. CA JARS z/OS data alone can be converted to the JR70CBF/CB2 format, providing the ability to report on the reformatted history file records via the Wizard Report Writer, including individual forms segment records that are created during the conversion process.

A CA JARS Wizard conversion program, JARSCONV, converts z/OS level-7 history file records into the format mapped by these glossaries. In this conversion process, the CAWGJR70 glossary maps the input Level-7 history files and is specified as input. The JR70CBF glossary is specified as output. The JARSCONV output files in JR70CBF format may then be concatenated with CA JARS VSE records in JD60VSE or JD62VSE format, using the JR70CBF or JR70CB2 glossary, respectively, for combined z/OS and VSE reporting. Or they may be used alone for z/OS reporting. JARSCONV is distributed as a member of CAJROPTN.

The JR70CBF/JR70CB2 glossaries contain the CA JARS VSE related record descriptions found in the VSE glossaries JD60VSE/JD62VSE plus two record descriptions that pertain to CA JARS z/OS: a reformatted history record and a record constructed from history record SYSOUT segments. Because of the large number of record descriptions, the JR70CBF/JR70CB2 glossaries are not presented in detail in this appendix. The Print Glossary Utility Program, WIZPG, described in the CA JARS Wizard Reference Guide, can be used to print these record descriptions.

Index